UNIVERSIDADE DE LISBOA

FACULDADE DE CIÊNCIAS

DEPARTAMENTO DE INFORMÁTICA

Quanto valem os metadados?

Mestrado em Engenharia Informática
Engenharia de Software

Bruno Filipe Prudêncio Inácio

Dissertação orientada por:
Francisco José Moreira Couto
João Diogo Silva Ferreira

2016

Agradecimentos

Para o desenvolvimento desta tese foi necessário conciliar três ambientes: o ambiente familiar, o ambiente profissional e o ambiente académico. Sem a ajuda da minha família direta seria impossível reunir o tempo necessário para a construção dos múltiplos blocos que constituem o edifício desta tese, e por isso tenho de agradecer a sua compreensão, e paciência, pela minha ausência em momentos chave da vida familiar. Sem a colaboração, e aposta, do meu organismo patronal, o Instituto Hidrográfico (IH) e de um conjunto pessoas que dele fazem parte, a realização desta tese, e do próprio mestrado, estaria em causa. Agradeço, por isso, ao Conselho Cientifico do IH pela autorização dada à realização do meu mestrado, e ao meu coorientador nomeado toda a ajuda por ele concedida. Por fim, tenho naturalmente de agradecer todo apoio prestado pelo orientador e coorientador desta tese, respetivamente o prof. Francisco Couto e o prof. João Ferreira.

Ao meu filho Guilherme,

e à minha esposa Andreia.

Resumo

As atividades de investigação e desenvolvimento estão cada vez mais dependentes da partilha de informação. O volume de dados gerados ou consumidos assume valores cada vez maiores em muitas áreas científicas. No entanto, as metodologias desenvolvidas e implementadas, no sentido de aumentar a quantidade e qualidade dos dados partilhados, têm apresentado sérias dificuldades em cumprir o propósito de facilitar essa partilha. Até ao momento, o caminho seguido tem sido a utilização de meros repositórios públicos onde os dados gerados pelas investigações são depositados, mas que não implementam funcionalidades que facilitem a partilha e integração dos dados por outros investigadores, sendo portanto difícil extrair conhecimento de forma automática destes dados.

É necessária assim uma nova abordagem à forma como esta partilha é feita. Uma abordagem que permita que a informação possa ser organizada, caracterizada e atualizada de modo contínuo. Esse trabalho poderá ser feito pelo investigador, que acima de tudo conhece o domínio dos dados, mas também por curadores, que conhecem tanto o domínio como as práticas de partilha. A maior barreira para a implementação desta metodologia é assim humana, sendo a motivação para organizar, caraterizar semanticamente e atualizar os dados um dos pontos-chave. Nesta tese é assumido que através da implementação de um mecanismo que recompense a partilha e a integração dos metadados que descrevem os conjuntos de dados, de acordo com os princípios da *Web* Semântica, estaremos a promover e a intensificar a confiança e qualidade na partilha e integração dos mesmos, como passo essencial no avanço científico. Para tal, é necessário que esta qualidade de integração possa ser avaliada, e assim averiguar a utilidade dos metadados e, consequentemente, do conhecimento proporcionado pelos metadados na descoberta dos conjuntos de dados.

Esta tese teve como objetivo o desenvolvimento de uma ferramenta que permite a avaliação do nível de conhecimento proporcionado pelos metadados utilizados para a descrição de conjuntos de dados de um qualquer repositório científico, tendo em conta a qualidade da sua integração semântica com ontologias públicas, de acordo com a especificidade das anotações com referência a conceitos ontológicos, utilizados para descrição das suas propriedades, e da completude desta integração. Deste modo, foi apresentado um estudo onde estes dois critérios (especificidade e cobertura de anotações) foram propostos como medidas de qualidade de integração semântica de metadados, partindo da representação formal de ontologia como um grafo acíclico. Estas medidas foram implementadas e utilizadas pela ferramenta de modo a analisar a qualidade dos

metadados utilizados por um repositório real de dados científicos, e assim efetuar uma avaliação quantitativa da implementação específica da ferramenta. Os resultados obtidos permitiram concluir que a ferramenta implementou corretamente as medidas estudadas, na avaliação da qualidade dos metadados, e que existe de facto uma fraca aposta, sobretudo quantitativa, na descrição semântica dos metadados por parte dos investigadores.

Palavras-chave: Partilha de Informação, Integração de Informação, *Web* Semântica, Metadados, Ontologias

Abstract

Research and development activities are increasingly dependent on information sharing. The volume of generated or consumed data assumes increasing values in many scientific areas. However, the methodologies developed and implemented to increase the quantity and quality of shared data, have presented serious difficulties in fulfilling the purpose of facilitating such sharing. So far, the followed path has been to use public repositories, where data generated by the investigations is deposited, but which fail to implement features that facilitate data sharing and integration by other researchers, making it difficult to extract actual knowledge in an automatic way from the deposited data.

It is necessary to develop a new approach to how this data sharing is performed; an approach that allows the information to be organized, characterized and continuously updated. This work can be done by the investigator, who above all knows the field of data, but also by curators, who know both the domain and the data sharing practices. However the biggest barrier to the implementation of this methodology is human, and the motivation to organize, semantically characterize and update the datasets is one of the key points in the process. In this thesis it is assumed that through the implementation of a mechanism that rewards and recognizes sharing and integration of metadata describing the datasets, according to the principles of the Semantic Web, we will promote and intensify the confidence and quality of data sharing and integration as an essential step in scientific progress. To this end, it is necessary that the quality of integration can be assessed, and thus that we can determine the utility of the knowledge provided by metadata in dataset discovery.

The main work carried out by this thesis, was the development of a tool that allows the assessment of the level of knowledge provided by metadata used to describe datasets of any scientific repository, taking into account the quality of its semantic integration with public available ontologies, according to the specificity of annotations with reference to the ontological concepts used to describe its properties, and completeness, of this integration. Thus, a study was made where these two criteria (specificity and annotations coverage) have been proposed as semantic metadata integration quality. These measures were implemented and used by the tool to examine the quality of the metadata used in a real repository of scientific data to describe its datasets, and thus make a quantitative assessment of the tool's specific implementation. The results show that the tool correctly implements the measures studied in the evaluation of metadata quality and that there is indeed a lack of effort, especially with regards to the quantity, in semantic description of metadata by researchers.

Keywords: Data Sharing, Data Integration, Semantic Web, Metadata, Ontologies

Conteúdo

Anexos

Lista de Figuras

Lista de Tabelas

Capítulo 1

Introdução

Uma grande parte das áreas de investigação e desenvolvimento científico, senão mesmo todas, tornaram-se nos tempos mais recentes em ciências que produzem e consumem um volume monumental de dados, de múltiplas fontes e em vários formatos [1]. A partilha de informação tomou lugar de destaque e passou a ser uma etapa de extrema importância para o sucesso de qualquer investigação científica. Esta partilha permite, sobretudo, que futuros avanços científicos possam ser consolidados com base em trabalho anteriormente desenvolvido pelos próprios investigadores, assim como no trabalho realizado por outros investigadores. No entanto, para que possam ser partilhados, estes dados devem ser organizados, categorizados e analisados, o que resulta naturalmente em uma maior compreensão e potencial aquisição de conhecimento sobre os mesmos. A integração e partilha de informação são, assim, dois requisitos-chave para o avanço do conhecimento científico. Desta forma, é importante que estes dois processos passem a ser vistos como passos fundamentais de investigação em qualquer comunidade científica.

Este conceito de partilha de informação sobre investigações realizadas não é, no entanto, um conceito recente. Quando Galileu (1610) publicou no *Sidereus Nuncius* (geralmente traduzido como Mensageiro Sideral) as observações de Júpiter e das suas luas [2], realizadas com auxílio de um telescópio, estaria já a potenciar o avanço da ciência na medida em que descrevia a outros investigadores não apenas os resultados obtidos nas suas descobertas, mas a forma como estes resultados poderiam ser compreendidos, analisados e sobretudo replicados.

Em algumas áreas de investigação científica, como a biomédica, p. ex., a necessidade de implementação de processos de partilha e reutilização de dados de investigação encontra-se já num estado bem esclarecido. Nesta área é terreno comum recorrer a aglomerados massivos de dados que necessitam de ser utilizados em múltiplas ações de investigação. Estes aglomerados surgem dos avanços recentes na área das tecnologias de informação, que permitiram o desenvolvimento de novos meios de captura de informação

e persistência de dados, p. ex., o registo eletrónico de saúde de um paciente ou a utilização de meios móveis para a recolha em tempo-real de informação de saúde [3].

A *Innovative Medicines Initiative* (IMI), uma parceria público-privada entre a Comissão Europeia e a associação da indústria farmacêutica europeia, prevê a criação de plataformas de salvaguarda e partilha de dados de modo a estabelecer uma base de dados que permita a partilha aberta de informação entre os seus membros, que tenham em conta a avaliação da qualidade dos metadados [4]. Estas plataformas têm como objetivo melhorar a saúde através da aceleração do desenvolvimento e acesso dos pacientes a medicamentos inovadores, de acordo com [1].

Existem, no entanto, inúmeras barreiras à correta partilha de informação, como sejam a inexistência de meios tecnológicos realmente adequados ao domínio de conhecimento científico em causa, a especificidade da informação de cada área de investigação ou até o fator humano na partilha de dados, talvez um dos fatores dos mais importantes. Ao longo do tempo têm vindo a ser desenvolvidos vários tipos de repositórios de dados [5], com diferentes paradigmas de implementação, com o objetivo comum de oferecer aos investigadores e respetivas universidades, laboratórios ou organizações um meio não só de salvaguarda de informação, gerada durante as suas ações de investigação, mas também de partilha de dados para investigações futuras dos próprios ou de outros investigadores. Repositórios como o *BioMart*, que apresenta uma interface único sobre múltiplas e distintas fontes de informação científica [6], o *Databib* (ou atualmente o *Re3data2*), um diretório de repositórios de dados científicos revistos manualmente de várias disciplinas académicas [7], o *LabKey*, que permite a integração de dados de múltiplas experiências num único ponto de modo a que possam ser analisados e partilhados não apenas os dados em bruto como o resultado das análises efetuadas [8], e o *SDCube*, um repositório de dados em bruto e respetivos metadados que combina o *Extensible Markup Language* (XML) e o *Hierarchical Data Format* (HDF) através de uma metáfora de cubo [9].

Apesar do esforço despendido no desenvolvimento de meios tecnológicos, e de abstração da complexidade de gestão dos dados utilizados pelos investigadores, a maior dificuldade na implementação de um modelo comum de partilha e integração de informação é social, por natureza [3]. Esta dificuldade surge, sobretudo, pelas atitudes e preocupações que os investigadores revelam sobre a integração e partilha dos seus próprios dados. E neste âmbito surge um conjunto de fatores que naturalmente influenciam o comportamento de cada investigador. Existem aqueles que revelam falta de destreza na utilização dos meios tecnológicos à sua disposição, p. ex., na identificação do repositório indicado e de que forma o utilizar. Existem aqueles que consideram simplesmente que a partilha de informação não é relevante para o seu trabalho. Existem aqueles que consideram a partilha relevante e necessária mas optam por não a fazer por

ser demasiado dispendiosa do ponto de vista do trabalho envolvido. Existem até aqueles que receiam que a partilha possa potenciar situações de abuso de utilização por parte de outros, sem o devido reconhecimento do mérito, ou receiam simplesmente uma outra qualquer nova situação [3].

Torna-se assim necessário a implementação de um paradigma diferente de partilha de informação através do qual seja possível o reconhecimento e a recompensa de quem a produziu e dedicou o tempo necessário à sua organização e caracterização, assim como na colaboração contínua de atualização dos repositórios de dados [1]. Este paradigma tem necessariamente de assentar sobre a caracterização dos conjuntos de dados recolhidos durante as ações de investigação, através do mapeamento entre conceitos utilizados na sua descrição e conceitos presentes em vocabulários de descrição de domínios de conhecimento, aceites e reconhecidos pela comunidade. Esta metodologia potencia não só a identificação clara (sem ambiguidade) dos metadados associados aos dados, como também o cálculo de algumas variáveis, como a especificidade e a distinção dos conceitos utilizados para na descrição das suas propriedades.

Os metadados têm assim de assumir um papel ainda mais reforçado e importante, na integração e partilha de conhecimento, de modo a potenciar o avanço científico. Este avanço apenas se torna possível tendo em conta a qualidade desse conhecimento e a utilidade dos metadados utilizados para o descrever. Esta utilidade, ou qualidade, é proporcional ao esforço colocado na sua construção, que deve ser feita da forma mais precisa e específica possível, do ponto de vista da qualidade do conhecimento proporcionado pelos conceitos utilizados na descrição das propriedades dos dados, que os metadados acompanham, tendo em conta a abrangência de utilização dos mesmos. No seu principal papel, os metadados devem permitir que o conhecimento que descrevem possa ser eficazmente descoberto, compreendido, integrado e utilizado por vários meios não só humanos, mas sobretudo computacionais, p. ex., navegadores de *Internet*, especialmente desenhados para navegar através de uma rede semântica de informação, como o *Falcons* [10], motores de indexação e pesquisa semântica, como o Watson [11], ou processos de *Data Mining* baseados na incorporação de domínios de conhecimento na descrição de dados [12].

Para além de serem considerados meramente um meio para um fim, uma mera ferramenta para gestão de outras mais-valias digitais, os metadados devem então ser considerados como um passo crucial no avanço científico. O esforço e tempo despendido na sua criação e manutenção são investimentos significativos por parte dos investigadores. Através do reconhecimento e recompensa é até possível considerar um retorno de investimento (ROI) do ponto de vista dos metadados [13], não apenas para os investigadores mas também para universidades, laboratórios ou organizações.

1.1 Problema

A correta integração e partilha de informação científica do ponto de vista semântico, para lá da sua mera inclusão em repositórios de dados, envolve a caracterização dos conjuntos de dados com o auxílio de metadados e do mapeamento dos conceitos utilizados na sua descrição a domínios de conhecimento públicos, reconhecidos pela comunidade de investigadores. Estes facilitam a descoberta de informação relevante acerca dos conjuntos de dados não apenas por pessoas, mas sobretudo por meios computacionais. No âmbito do paradigma da *Web* Semântica, estabelecido por [14], descrito na Secção 2.1, a identificação dos termos que constituem os metadados deverá ser feita através da utilização de apontadores para recursos externos.

Estes recursos são o vocabulário de descrição desses metadados, que permitem a identificação de modo não ambíguo dos conceitos que os descrevem, no contexto de um domínio de conhecimento particular. Estes vocabulários são por vezes adaptados, pelas comunidades científicas como uma descrição correta e útil do domínio de conhecimento normalizado e de acordo com a sua complexidade poderão ser designados de ontologias (uma ontologia pode ser descrita informalmente como um vocabulário de termos e alguma especificação sobre o seu significado). Isto permite que a descrição de recursos possa ser normalizada e universalmente aceite, constituída e acedida, o que facilita a integração dos dados e posterior partilha, potenciando assim a sua utilidade e o avanço cientifico.

No entanto, esta integração de dados é geralmente feita pelos próprios autores, apesar de alguns avanços tecnológicos mais recentes, como as ferramentas de gestão, anotação e partilha de metadados apresentadas na Secção 2.4. Dada a especificidade das áreas de investigação, e em muitos casos dado também o desconhecimento do vocabulário de descrição do domínio de conhecimento a utilizar, a caracterização dos conjuntos de dados com recurso ao paradigma da *Web* Semântica torna-se um processo complexo e demorado para os investigadores, especialmente se for executado sem o auxílio de mecanismos automáticos [3]. Sem uma obrigatoriedade implícita da correta integração entre conjunto de dados e respetivos metadados, torna-se então difícil encontrar publicações de dados corretamente anotadas, do ponto de vista semântico, nos vários repositórios disponíveis, não passando estes de simples silos de informação dos quais dificilmente se extrai conhecimento [1].

Do ponto de vista do novo paradigma de partilha de informação mencionado anteriormente, baseado no reconhecimento e recompensa de quem mais esforços dedica à correta descrição dos dados que publica, torna-se necessário definir o conceito de qualidade de integração semântica dos metadados que lhes estão associados. Tendo em conta que a criação destes deva ser feita de acordo com os princípios da *Web* Semântica,

onde a interoperabilidade entre os recursos identificados é a chave, e atingida exclusivamente pelo mapeamento entre eles, a sua qualidade deverá ser tanto maior quanto o número e nível de detalhe das anotações a recursos externos, que os seus elementos apresentem.

A motivação, e o sentido de utilidade, com a qual os investigadores efetuam este mapeamento são dois problemas que afetam seriamente o processo de integração e partilha de dados. Por um lado, é necessário convencer os investigadores a despender parte do seu tempo à organização, catalogação e descrição semântica dos seus dados. Por outro lado, é necessário que esta descrição possa ser o mais útil possível para quem os queira utilizar, do ponto de vista do conhecimento potenciado pela escolha de conceitos que sejam parte constituinte de vocabulários de descrição do domínio de conhecimento, normalizados e reconhecidos pela comunidade de investigadores.

Quanto maior for a utilidade, ou facilidade, dos metadados em dar a conhecer os dados que caraterizam, a quem os pretenda utilizar, maior terá sido o esforço empregue na qualidade da sua descrição semântica. Esta qualidade de integração semântica dos metadados pode ser interpretada por duas vertentes, em conjunto: (*i*) pelo número de anotações a conceitos, ou a sua abrangência, presentes em vocabulários externos, utilizadas na descrição dos metadados; (*ii*) pelo detalhe de conhecimento, ou de especificidade do seu significado, proporcionado por cada um desses conceitos na sua relação com outros, no vocabulário do domínio de conhecimento a que pertence. Um conceito que não seja utilizado por outros para a descrição dos próprios, i.e., sem ligações ascendentes, é um conceito cujo significado pode ser melhor especificado, do ponto de vista do conhecimento que proporciona. Um conceito que não utilize qualquer outro para a sua descrição, i.e., sem ligações descendentes, é um conceito cujo conhecimento proporcionado não pode ser melhor especificado.

No entanto, não existem ferramentas tecnológicas capazes de avaliar a qualidade de integração semântica dos metadados, tendo em conta as vertentes referidas, logo a sua utilidade para a compreensão e utilização dos dados que descrevem. Existem inúmeros repositórios de estudos científicos, como aqueles referidos anteriormente, que permitem a inclusão de metadados sem efetuarem uma avaliação da sua integração semântica. Existem também ferramentas especializadas na construção, categorização e gestão de metadados (ver Secção 2.4), com uma forte componente de anotação semântica, no entanto, sem nunca efetuarem uma avaliação concreta quer do número de anotações utilizadas, quer da especificidade do significado a cada conceito.

1.2 Objetivos

Esta tese tem como objetivo desenvolver e validar mecanismos capazes de avaliar a qualidade de integração semântica dos metadados, de um conjunto de dados, do ponto de vista da sua integração com vocabulários externos, i.e., a avaliação do nível de conhecimento proporcionado pelos conceitos utilizados para a anotação das propriedades de descrição de metadados, assim como da completude ou abrangência da sua utilização. Estes conceitos estão englobados em domínios de conhecimento de acesso público, ou ontologias, aceites pela comunidade a que pertencem como descrições precisas e abrangentes do conhecimento científico relativo a essa comunidade. O nível de conhecimento é proporcional à especificidade semântica dos conceitos utilizados. Esta é calculada tendo em conta a posição do conceito no vocabulário de referência, indicado pela anotação de um termo utilizado na descrição dos metadados (ver Secção 3.1).

Através desta avaliação da qualidade de integração semântica dos metadados utilizados é possível medir o nível de integração semântica dos mesmos e o valor associado à partilha de um determinado conjunto de dados. Este valor poderá ser a base para um novo mecanismo de recompensa e reconhecimento [1], que tem como principal elemento uma moeda virtual. Para cumprir com o este objetivo será necessário:

1. Realizar um estudo de metodologias: (*i*) de avaliação daquilo que se considera ser a especificidade de uma anotação, do ponto de vista da ontologia a que o conceito utilizado pertence; (*ii*) de cálculo de cobertura das anotações perante o conjunto de ligações a conceitos de ontologias, utilizados na descrição dos metadados.

2. Desenvolver uma ferramenta que permita a aplicação destas metodologias na avaliação de metadados que caraterizam um conjunto de dados de um qualquer repositório, quer por um utilizador humano, quer por meios computacionais, que servirá também para validar as metodologias do ponto anterior.

3. Efetuar a avaliação da implementação da ferramenta através da sua aplicação a um caso real, e posterior recolha e análise dos resultados quantitativos.

1.3 Metodologia

O desenvolvimento desta tese foi elaborado ao longo de três etapas: (*i*) estudo do estado da arte e desenho preliminar das contribuições a implementar; (*ii*) implementação das contribuições; (*iii*) desenho e implementação de testes e análise de resultados. A primeira etapa envolveu o estudo da temática sobre a qual esta tese recai. Para tal, foi feita uma investigação sobre o que é a *Web* Semântica, quais as suas origens, que iniciativas existem para além deste tema, em que estado se encontra o desenvolvimento e implementação do

conceito de *Web* Semântica. Esta etapa envolveu também a investigação sobre o problema apresentado e qual a melhor forma de elaborar a contribuição da tese. Neste âmbito, foi estudada a metodologia que melhor se adequa ao cumprimento dos objetivos da tese, na avaliação da qualidade de integração semântica dos metadados associados aos dados gerados durante as ações de investigação. Foram também analisados estudos anteriores sobre o tema de avaliação semântica, assim como possíveis repositórios de dados científicos que permitissem a definição de um caso de estudo, para validação das medidas de qualidade de integração desenvolvidas.

Uma vez compreendido o problema e encontrada uma solução sobre a metodologia de avaliação semântica dos metadados, foi feita uma análise sobre a arquitetura de desenho da plataforma de avaliação. Um dos desafios colocados a esta análise foi o desenhar de uma solução que permitisse a avaliação de todos conceitos presentes em um ficheiro de metadados num período de tempo que fosse considerado aceitável pelo utilizador da plataforma de análise e avaliação. A solução encontrada recaiu sobre a utilização de um motor local de base de dados, e por esse motivo foi necessário novo estudo sobre a metodologia de conversão a adotar, do formato original dos vocabulários para um modelo relacional, de modo a concluir a sua implementação.

A segunda etapa envolveu a implementação de um ambiente de desenvolvimento que permitisse edificar a arquitetura definida, em todos os seus componentes. Para tal, foi necessário elaborar uma lista de tecnologias passíveis de serem utilizadas para o efeito. A criação desta lista envolveu a investigação das caraterísticas de cada tecnologia e meios de implementação necessários. Uma vez identificadas as tecnologias que poderiam contribuir para a solução, o ambiente de desenvolvimento foi implementado e construída a contribuição da tese.

Na terceira etapa foi implementado um ambiente de testes, que permitiu avaliar as capacidades da solução encontrada. Estas capacidades foram analisadas em duas vertentes: (*i*) teste das metodologias escolhidas para avaliação de metadados; (*ii*) teste da plataforma de avaliação global da qualidade de integração semântica dos metadados associados a conjuntos de dados metabólicos de um repositório do *European Bioinformatics Institute* (EMBL-EBI), chamado *MetaboLights* [15]. Na primeira vertente, foi utilizado um conjunto de ontologias, que variam em tamanho e em profundidade de relações dos seus termos, de onde foi recolhida uma amostra. Estes foram avaliados pelo procedimento desenvolvido e os seus resultados comparados com os resultados de estudos anteriores. Na segunda vertente, foi implementado um caso de estudo sobre o qual foi testada a solução desenvolvida. Os resultados obtidos foram comparados com os resultados de estudos anteriores e conferidos manualmente.

1.4 Contribuições

As contribuições desta tese para a resolução do problema enunciado, foram estabelecidas do seguinte modo: (*i*) elaboração de um artigo técnico no qual é apresentado um estudo sobre medidas de qualidade de integração semântica dos metadados, onde se encontram explicadas as fórmulas para o cálculo do valor de especificidade de um conceito perante o seu vocabulário de referência e para o cálculo do nível de cobertura de anotações utilizadas na descrição dos metadados por anotações com referência a termos de vocabulários externos; e a (*ii*) construção de uma plataforma, que dado um ficheiro de metadados avalia a qualidade da sua integração semântica, assim como uma posterior análise de resultados de âmbito geral, tendo em conta as medidas descritas no artigo anterior. Esta ferramenta representa uma generalização do modelo desenvolvido num estudo anterior numa disciplina do Mestrado em Bioinformática e Biologia Computacional [16], na medida em que deverá implementar uma versão mais abstrata do modelo utilizado para cálculo das medidas de especificidade e cobertura de termos, permitindo assim a análise de metadados de um qualquer repositório, sem que esteja restrita a um único modelo de metadados; (*iii*) aplicação da ferramenta a um repositório real, onde se encontrem salvaguardados os dados de múltiplas experiências, a partir do qual seja possível medir o nível de integração semântica dos metadados que acompanham cada uma dessas experiências.

O resultado do estudo sobre medidas de qualidade de integração semântica é parte integrante e fundamental da funcionalidade da plataforma desenvolvida. As fórmulas encontradas são utilizadas na construção de procedimentos, baseados numa representação relacional de ontologias, que calculam um valor normalizado de especificidade de um conceito utilizado como descritor de metadados.

A plataforma é formada por quatro componentes ou camadas: (*i*) o repositório de ontologias, que garante a salvaguarda prévia de ontologias que podem ser utilizadas para avaliação dos metadados submetidos; (*ii*) o motor de análise e avaliação que é o responsável pela recolha e medição de anotações utilizadas nos metadados; (*iii*) a interface computacional, que permite que a solução possa ser ela própria integrada em outras soluções computacionais através da *Internet*; (*iv*) a interface de utilizador, que representa o ponto de ligação entre o utilizador e a ferramenta, através do qual é possível, entre outras funções, a submissão dos metadados a avaliar.

1.5 Resultados

Os resultados apresentados por esta tese foram divididos em dois grupos: (*i*) os resultados da implementação das medidas de avaliação de qualidade de integração semântica dos metadados (especificidade e cobertura), através de um conjunto de procedimentos

desenvolvidos diretamente sobre o modelo relacional definido como solução; (*ii*) o resultado da implementação da plataforma de análise e avaliação de metadados, que utiliza o conjunto de procedimentos anterior, na avaliação de todos os ficheiros de metadados presentes num repositório real de dados experimentais.

O primeiro grupo de resultados confirma que a implementação das medidas de avaliação de especificidade está de acordo com aquilo que são os objetivos do seu estudo. No segundo grupo de resultados, e dada a potencial diversidade de ontologias que podem ser utilizadas para anotação de metadados, foi necessária a apresentação de um caso de estudo. Assim o repositório utilizado para avaliação de resultados obtidos, através da nova ferramenta, foi aquele utilizado por um estudo anterior [16], o repositório *MetaboLights* [15]. Este é um repositório recente de experiências na área metabológica e informação derivada, suportado pelo EMBL-EBI com uma forte participação da comunidade científica. Os resultados obtidos neste segundo grupo revelam por um lado, que a plataforma apresenta uma maior completude e precisão na avaliação dos metadados, do que aquela apresentada pelo estudo descrito em [16], pois não só consegue encontrar e extrair um maior número de termos, como apresenta um valor mais real da especificidade das anotações utilizadas, na descrição dos metadados. Por outro e com base em um procedimento de verificação manual, os resultados obtidos revelam que o cálculo da especificidade das anotações está de acordo com as medidas de avaliação de qualidade de integração semântica dos metadados, apresentadas na Secção 3.1

Os resultados revelam ainda, de um modo geral, uma tendência vincada no repositório *MetaboLights* para a fraca integração semântica de metadados, identificada no problema da tese. Estes sugerem que existe uma ausência de anotações a conceitos de vocabulários externos, para anotação dos termos utilizados na descrição dos metadados. No entanto, os conceitos utilizados revelam uma alta média de especificidades semântica, i.e., os conceitos escolhidos para anotação potenciam um valor alto de conhecimento sobre a caraterização dos metadados a quem os pretenda utilizar.

1.6 Planeamento

A Tabela 1-1 apresenta o planeamento original definido para esta tese. Este foi cumprido na íntegra em todas as suas fases. No entanto, existem etapas que deveriam ter tido mais tempo de execução, p. ex., a etapa de desenvolvimento do motor genérico de análise. O tempo atribuído revelou-se curto para o nível de trabalho que representou o desenvolvimento do principal segmento da contribuição desta tese.

Tabela 1-1 - Planeamento de tarefas da tese.

Início	Fim	Duração	Tarefa
09/2015	10/2015	5 Semanas	**Trabalho de investigação:** • Pesquisa e leitura de vários artigos sobre o tema; • Análise do estudo sobre *Knowledge Rating* aplicado ao repositório *MetaboLights*; • Análise dos algoritmos desenvolvidos no estudo; • Análise do repositório *MetaboLights*.
10/2015	10/2015	2 Semanas	**Elaboração do relatório preliminar**
10/2015	02/2016	12 Semanas	**Desenvolvimento do motor genérico de análise:** • Definição e análise de requisitos; • Escolha de arquitetura e desenho detalhado; • Codificação do sistema; • Elaboração de testes.
02/2016	03/2016	6 Semanas	**Desenvolvimento da interface:** • Especificação e desenho detalhado; • Codificação da interface; • Elaboração de testes.
03/2016	04/2016	5 Semanas	**Recolha de informação e avaliação do sistema através de um *Case Study* (*MetaboLights*)**
04/2016	06/2016	5 Semanas	**Escrita da tese**
05/2016	06/2016	5 Semanas	**Escrita do artigo a publicar**

1.7 Estrutura do documento

Esta tese está organizada da seguinte forma:

- **Capítulo 1** – Introdução e apresentação do problema e objetivos que se pretendem cumprir com a realização desta tese.
- **Capítulo 2** – Trabalho relacionado – Apresenta os conceitos necessários para a compreensão do paradigma sobre o qual se baseia esta tese, assim como a análise de várias soluções já existentes na área de integração semântica.
- **Capítulo 3** – Trabalho desenvolvido – Descreve o estudo sobre as medidas adotadas para avaliação da qualidade de integração semântica dos metadados, assim como o processo de desenvolvimento e implementação da plataforma de análise e avaliação de metadados.
- **Capítulo 4** – Resultados – Apresenta os resultados obtidos para o procedimento de implementação das medidas de avaliação semântica, os resultados obtidos no caso de estudo sobre o motor de análise e avaliação de metadados, assim como os resultados obtidos de um questionário de usabilidade da solução.
- **Capítulo 5** – Conclusão – Descreve a conclusão da tese e trabalho futuro.
- **Anexos A - O** – Contêm informação necessária à compreensão da arquitetura da plataforma desenhada e implementada como contribuição desta tese, assim como informação necessária à validação dos resultados obtidos durante os testes efetuados à plataforma, para além do questionário enviado a potenciais utilizadores da plataforma.

Capítulo 2

Trabalho relacionado

2.1 Web Semântica

O paradigma no qual se baseia o estudo desta tese é o paradigma de *Web* Semântica. Este foi introduzido por Tim Berners-Lee *(*2001), e comtempla a *Internet* como uma rede de dados e não apenas como uma rede de documentos e hipertexto [14] (páginas de *Internet*). Na *Internet* de hipertexto, maioritariamente desenvolvida em HTML, as relações entre documentos são meras hiperligações que o utilizador utiliza para navegar entre as páginas, sem qualquer tipo de conhecimento intrínseco ou significado formal, sobre o contexto das mesmas. A *Web* Semântica, por seu lado, apresenta uma rede de dados onde cada nó apresenta um domínio distinto, mas onde o fator de maior relevância é a relação que cada elemento tem com os restantes. Neste paradigma, as hiperligações entre documentos de texto são substituídas por ligações, não só entre os vários conjuntos de dados, como também entre os dados e conceitos descritos em ontologias.

Esta rede de interligações constitui uma abstração dos domínios de conhecimento do mundo real, dos seus elementos e das relações existentes entre si. E são estas que de fato importa capturar, pois é a partir da interligação entre elementos que emerge um conhecimento intrínseco sobre a caracterização de cada um deles. Por exemplo, na criação da nossa rede social de conhecimentos temos natural tendência de criar ligações com pessoas com as quais temos alguém que conhecemos em comum, i.e., temos tendência a obter conhecimento sobre alguém através da análise das suas relações sociais com outras pessoas.

Tim Berners-Lee (2006) estabeleceu na sua visão de *Web* Semântica através de um grupo de três "regras" [17], que não sendo obrigatórias, devem ser seguidas de modo a obter um elevado grau de interligação, ou de relacionamento, entre elementos de dados existentes na *Internet*: (*i*) identificar elementos, objetos ou conceitos através de um identificador único, permitindo assim que a sua identificação seja única no universo da *Internet*; (*ii*) utilizar o esquema do padrão *Hypertext Transfer Protocol* (HTTP) e respetivas capacidades, detalhado na Secção 2.5.5, para a construção de endereçamentos únicos, i.e., não desenvolver e utilizar quaisquer outros esquemas ou sub-esquemas de endereçamento; (*iii*) sempre que um destes endereços for procurado, deve ser fornecida informação útil sobre o mesmo, i.e., sempre que o elemento for visitado fornece ao visitante informação sobre si, através de outras ligações a outros elementos.

2.1.1 Elementos-chave

O paradigma de *Web* Semântica foi desenvolvido em torno de um conjunto de elementos-chave [18]. A informação sobre o elemento visitado deve ser dada através de tecnologias bem estabelecidas e por isso bem conhecidas na comunidade, como são (*i*) a plataforma *Resource Description Framework* (RDF) ou (*ii*) a linguagem *Protocol and RDF Query Language* (SPARQL); (*iii*) a *Uniform Resource Identifiers* (URI,) e (*iv*) ao protocolo HTTP.

O RDF [19] faz parte das recomendações do *World Wide Web Consortium* (W3C) [20] e estabelece um modelo formal de interligação de informação na *Internet*, através do qual é possível estabelecer relações entre modelos de dados. Estruturalmente forma um grafo, onde as arestas representam ligações fortemente caracterizadas e os nós representam os objetos. O SPARQL faz também parte das recomendações do W3C e especifica uma linguagem de pesquisa sobre o modelo formal definido pelo RDF [21]. O recurso a estas ferramentas tem como objetivo criar uma rede de conhecimento onde cada ligação estabelece uma relação semântica entre elementos, i.e., onde uma ligação não é apenas um caminho que deva ser seguido na procura de conhecimento, mas uma propriedade com significado em si mesma, na identificação do elemento com recurso a outro.

O URI é um identificador único na *Internet* de recursos ou entidades que podem ser consultados publicamente, através da *Internet* [22]. Estes obedecem a um esquema HTTP (http:// ou https://) de endereçamento, que assenta numa hierarquia pública e descentralizada de nomes na *Internet*, o *Domain Name System* (DNS), para a tradução da sua forma literal num endereço que possa ser compreendido de modo computacional, o endereço de *Internet Protocol* (IP) [23]. Por exemplo, o URI *http://www.ebi.ac.uk/efo/EFO_0000400* identifica singularmente o termo "*diabetes mellitus*" na ontologia *Experimental Factor Ontology* (EFO) na *Internet*, i.e., o mesmo endereço não pode corresponder a dois conceitos diferentes.

O HTTP é o meio utilizado para leitura do conteúdo para o qual o URI aponta, i.e., através deste protocolo de comunicação é possível estabelecer uma ligação através da *Internet* e obter o conteúdo ou a descrição da entidade que o URI identifica. O HTTP e o URI são elementos estruturais na construção da rede semântica, mas sem capacidade, no entanto, de expressarem por si próprios maior conhecimento do que aquele que uma simples ligação a uma entidade consiga revelar. O RDF incorpora o URI na definição de uma rede semântica na medida em que fornece um modelo de dados, em modo de grafo, que possibilita que as ligações entre entidades possam ser enquadradas de forma a contribuírem para o seu significado de forma computacional, e assim permitirem dar aos

sistemas automáticos a capacidade de trabalhar com o significado dos objetos e não só com a sua representação digital.

O modelo RDF assenta em três conceitos-chave: (*i*) sujeito, (*ii*) predicado e (*iii*) objeto. Estes formam aquilo a que se chama "um triplo RDF" [19], onde o sujeito e o objeto identificam entidades que se pretende ver relacionadas, através de um predicado. Por exemplo, para expressar literalmente que duas pessoas, A e B, se conhecem de algum modo simplesmente afirmamos que a pessoa A conhece a pessoa B. Através do modelo RDF traduzimos que a pessoa A (sujeito) conhece (predicado) a pessoa B (objeto). Se consideramos que todos os elementos desta expressão são entidades que podem ser identificadas de modo único e consultadas na *Internet*, aplicando assim os princípios enumerados por Tim Berners-Lee, i.e., a cada uma das entidades corresponde um URI, podemos estabelecer uma ligação entre os recursos. Se consideramos que ambos os sujeitos A e B podem ser identificados através de URI, respetivamente: http://organizacaoA/pessoas/pessoaA e http://organizacaoB/pessoas/pessoaB; o triplo RDF anterior pode ser rescrito estabelecendo o predicado da seguinte forma:

Sujeito: http://organizacaoA/pessoas/pessoaA
Predicado: http://xmlns.com/foaf/spec/#term_knows
Objeto: http://organizacaoB/pessoas/pessoaB

O URI utilizado no predicado deste exemplo identifica um termo num vocabulário público, o *Friend Of A Friend* (FOAF), que define um domínio de conhecimento sobre a descrição de pessoas, as suas relações e atividades em relação a outras pessoas ou objetos [24]. Quer o sujeito, quer o objeto, podem por sua vez serem descritos por quaisquer outras relações, que envolvam quaisquer outros vocabulários públicos disponíveis na *Internet*. Estes vocabulários são eles próprios modelos de dados que relacionam termos e propriedades de um domínio de interesse particular, através de triplos RDF. Por sua vez, qualquer vocabulário pode relacionar ou integrar termos de outros vocabulários através do mesmo mecanismo, i.e., através da declaração de triplos RDF, onde os termos de outros vocabulários são identificados através de URI, e desta forma integrar na sua composição elementos de outros domínios de conhecimento.

Resource Description Framework Schemas (RDFS) [25] e *Web Ontology Language* (OWL) [26] são linguagens baseadas em RDF que são utilizadas para descrever, não as relações entre os objetos, mas sim quais os tipos de relações que podem existir entre eles. São assim formas de descrever factos genéricos e abstratos acerca de conceitos de um determinado domínio de conhecimento e providenciam, portanto, uma conceptualização da realidade. Na Secção 2.2 encontra-se descrito com mais detalhe como é que estas

conceptualizações (chamadas ontologias) são utilizadas. Neste exemplo, o vocabulário FOAF faz uso de RDFS para exprimir factos como o tipo de elementos que podem ser relacionados com o predicado *#term_knows*.

Através dos princípios da *Web* Semântica, em particular com recurso ao modelo formal RDF, é possível estabelecer assim uma rede onde vários elementos podem ser integrados, a partir de vários domínios de conhecimento e sem qualquer necessidade de conversão de formato, de modo a inferir outro nível de conhecimento, que de outra maneira continuaria impossível de estabelecer. Acresce que esta visão é implementada sobre a infraestrutura de *Internet* existente, através dos mecanismos de navegação do protocolo HTTP e identificação do URI. Possibilita assim um paradigma de integração de dados ao nível da dimensão da *Internet*, impossível de ser concebida até ao momento, onde os múltiplos domínios de conhecimento eram meras ilhas de informação, particularmente focado no significado, ou semântica, e contexto da informação.

Se a *Internet* dos documentos foi inicialmente concebida para a anotação e partilha de informação entre pessoas, a *Web* Semântica apresenta-se como uma *Internet* de dados que pode ser facilmente interpretada por meios computacionais, onde através da pesquisa da rede se consegue navegar sobre conjuntos de dados díspares e obter um grau de informação que de outro modo seria impossível obter, ou apenas com um enorme esforço. A inferência de informação, através do raciocínio da mesma com base na sua representação formal, é talvez um dos maiores ganhos da implementação deste paradigma, pois permite estabelecer relações que simplesmente não existiam à partida e com isso obter um ainda maior conhecimento semântico sobre os elementos que constituem a rede, tudo isto feito com um poder computacional cada vez maior.

2.1.2 Linked Data

Através da *Web* Semântica um novo conceito ganhou forma para publicação de dados na *Internet*. Este recebeu o nome de *Linked Data* [17], por Tim Berners-Lee (2006), suportado pelas tecnologias HTTP, RDF e URI, traduz um conjunto de melhores práticas para a publicação e interligação de dados estruturados, com recurso a ligações semânticas entre fontes de informação díspares. Ligações essas que constituem uma rede de dados relacionados entre si, de tal modo que pode ser navegada e compreendida sobretudo por meios computacionais, de modo a obter uma representação formal do conhecimento.

Tendo em conta este conceito Tim Berners-Lee (2009) voltou a reestabelecer os princípios de *Web* Semântica, numa conferência [27]: (*i*) todos os conceitos deveram ter um nome, um URI, que se inicie por *http://*; (*ii*) na resolução destes nomes, deverá ser obtida informação numa representação formal; (*iii*) o conteúdo da resposta não deverá

conter meras descrições sobre o conceito, mas sim relações entre este e outros conceitos identificados por URI.

Para lá das ligações estáticas entre documentos presentes na *Internet* de hipertexto, i.e., de ligações sem a capacidade de expressar outro tipo de relação entre documentos que não seja apenas delinear um caminho que o navegador deva percorrer, o conceito de *Linked Data* permite que ligações entre documentos arbitrários de RDF, que caracterizam grupos de dados identificados por URI, possam ser por sua vez caracterizadas por vocabulários externos, de modo a construir um conhecimento formal que pode ser obtido através da navegação das várias ligações existentes.

A maior expressão da implementação do conceito *Linked Data* é o *Linking Open Data Project* [28], iniciado em 2007 e suportado pelo *Semantic Web Education and Outreach* (SWEO) *Interest Group*[1]. Este projeto tem como principal objetivo englobar o maior número de conjuntos de dados existentes, de cariz público, através da sua conversão para RDF de acordo com os princípios do conceito *Linked Data*, para posteriormente criar ligações semânticas entre eles. Todos estes grupos de dados são disponibilizados publicamente para que possam ser integrados em outros domínios de conhecimento. Integração e partilha de informação são assim os conceitos base que este projeto advoga, com base na visão da *Web* Semântica, de modo a que esta possa ser considerada a plataforma de integração de dados à escala da *Internet*, independente dos domínios de conhecimentos, ou ontologias, e focada na semântica, ou significado, e contexto da informação.

Também a nível europeu existem projetos que implementam o conceito de *Linked Data*. O *EU Open Data Portal* [29] é um projeto que tenta englobar informação proveniente de várias instituições, agências e outros organismos europeus através da sua interligação sobre o modelo formal de RDF com pesquisa suportada por SPARQL. Em Portugal, o projeto Dados.gov 2.0, previsto para 2016, pretende agregar informação produzida pela administração pública de modo a que possa ser integrada semanticamente com outros vocabulários e padrões utilizados em outros portais europeus [30].

2.1.3 Linked Data Cloud

O resultado da integração dos vários domínios de informação, do *Linking Open Data Project*, é a criação de um aglomerado publicamente acessível de domínios de dados interligados entre si através dos princípios de *Web* Semântica, a que se dá o nome de *Linked Data Cloud*, que pode ser livremente incrementada [31]. Esta tem um conteúdo diversificado por natureza, compreendendo dados acerca localizações geográficas,

[1] https://www.w3.org/blog/SWEO

pessoas, companhias, livros, publicações científicas, filmes, musica, televisão, programas rádio, genes, proteínas, drogas e testes clínicos, comunidades, dados estatísticos entre outros [18].

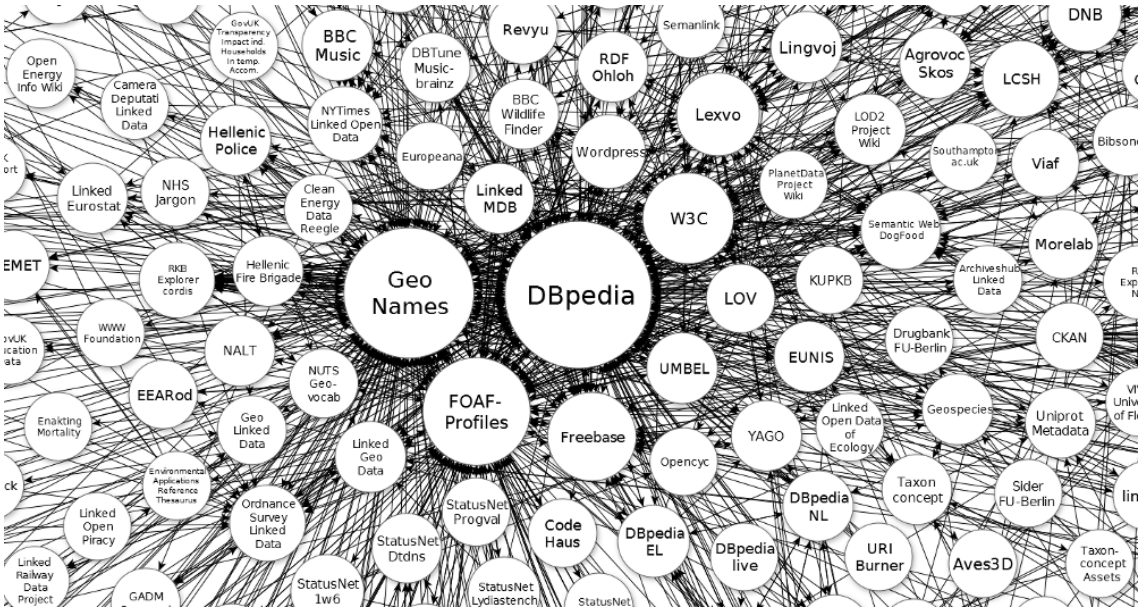


Figura 2-1 – Diagrama de ligações entre ontologias do Linked Open Data

O real tamanho desta estrutura apenas pode ser estimado, pois encontra-se em constante mudança. As estatísticas são recolhidas pela comunidade de apoio ao projeto, o *Linked Open Data* (LOD), que para além desta recolha mantém também o diagrama de interligação de todos os conjuntos de dados existentes, o LOD *cloud diagram* [32] (ilustrado na Figura 2-1). A informação estatística mais recente remonta a agosto de 2014 e aponta para a existência de 1014 conjuntos de dados, distribuídos por vários domínios de informação como governamentais, publicações e redes socais [33].

Da coleção de conjuntos de dados é de referir o *DBpedia* que de acordo com Tim Berners-Lee (2009) é um dos maiores esforços na direção do conceito *Linked Data* [34]. Este conjunto de dados é formado pela coleta de informação estruturada do projeto *Wikipedia*. Este conjunto de dados permite aos utilizadores inquirirem semanticamente as ligações e propriedades de todos os recursos existentes no *Wikipedia*, assim como interligações entre estes recursos e conjunto de dados de outros domínios de conhecimento [35]. Em outubro de 2015 o conjunto de dados da *DBpedia* tinha 6.2 milhões de entidades, das quais 5 milhões foram interligadas com ontologias consistentes. O conjunto era descrito formalmente por 1.1. mil milhões de triplos RDF.

2.2 Integração semântica dos metadados

A visão proposta para uma *Web* Semântica por Tim Berners-Lee *et al.* (2001) abriu portas a um novo paradigma de publicação e partilha de informação, o *Linked Data*, onde

documentos de texto (hipertexto) foram substituídos por conjuntos de dados [17]. A ligação entre os documentos, sem significado explícito contido em si própria, foi substituída por ligações que podem ser fortemente caraterizadas por vocabulários representantes de domínios de conhecimento, garantindo-lhes assim uma forte componente semântica, ou significado. Isto permitiu a abstração da problemática do formato dos dados na integração de conjuntos de dados, por parte da comunidade de investigadores, através da aplicação do modelo formal RDF sobre um conjunto comum de terminologias.

Tendo em conta que a investigação científica gera cada vez um maior número de dados em várias áreas e que, do ponto de vista do conhecimento obtido, o seu valor é tanto maior quanto maior é a facilidade de acesso e análise aos mesmos, a sua integração com outros conjuntos dados e partilha com a comunidade científica assume uma cada vez maior importância no sentido em que evita a criação de meros repositórios de informação científica, complementarmente "herméticos" do ponto de vista da sua compreensão pela comunidade científica, e até mesmo pelos próprios autores da informação [1]. A reutilização por meio da interligação da informação é assim um dos fatores mais importantes no avanço científico, mas para tal, é necessário que os dados gerados sejam (*i*) organizados, (*ii*) caraterizados e (*iii*) atualizados continuamente, de modo a que a sua compreensão e reutilização seja de facto possível. Um dos passos essenciais nesse processo é a descrição formal do conteúdo dos dados, i.e., a criação de metadados. Sem eles muito dificilmente a informação pode ser compreendida e reutilizada na anotação deve possuir.

Do ponto de vista da introdução do conceito de *Linked Data* na partilha de dados científicos, a criação dos metadados deverá ser feita de modo a que as descrições utilizadas na caraterização dos dados possam ser identificadas de modo único e não ambíguo, através de anotações a termos existentes em vocabulários externos, aceites pela comunidade científica, i.e., a descrição dos dados deve ser feita com recurso a um modelo formal e cada um dos termos utilizados deve ser anotado através de um endereçamento único (URI), que aponte para um recurso externo inserido numa ontologia reconhecida por todos os membros da comunidade, que caraterize o domínio de conhecimento no qual o termo está inserido. Isto permite que um conjunto de dados possa ser descrito por conceitos aceites e estabelecidos por todos os intervenientes nos processos de investigação, potenciando a sua integração e interoperabilidade na descoberta de novo conhecimento, assim como o avanço da ciência baseado em informação e conhecimento recolhidos por toda a comunidade de investigação.

Podemos então considerar os metadados como uma ponte semântica entre o conjunto de dados que eles descrevem e aqueles que os querem utilizar. Uma ponte que pode ser

percorrida e compreendida por meios computacionais, para além de humanos, na integração e partilha de informação, na pesquisa de dados ou reconhecimento de padrões. Uma ponte que pode ser avaliada de modo a quantificar o beneficio que oferece na integração dos dados que carateriza numa rede de conjuntos de dados, que pode ser computacionalmente interpretada. Do ponto de vista dos objetivos desta tese, que se baseiam na avaliação da qualidade de integração semântica dos metadados, i.e., da sua integração semântica com recurso a vocabulários externos, ou se quisermos da robustez desta ponte semântica, que acompanham a informação gerada por investigadores, importa enumerar os conceitos utilizados no processo de análise e avaliação dos mesmos:

Ontologia

Uma ontologia pode ser descrita informalmente como um vocabulário de termos e alguma especificação sobre o seu significado [36]. Estes termos são caraterizados por tipo e por propriedade, o seu significado é dado pelas interligações que descrevem com outros termos presentes no mesmo domínio de conhecimento, que a ontologia retrata, ou com outro domínio externo. Podem ser expressas em RDFS, OWL ou OWL2 (na especificação mais recente [37]), apesar de existirem outros formatos de menor utilização como o OBO *Flat File Format Specification* [38], p. ex.. se uma ontologia é aceite pela comunidade científica então a representação que esta apresenta do seu domínio de conhecimento passa a ser referência para todo o corpo de investigação. Por exemplo, a ontologia CHMO descreve métodos utilizados para a coleta de dados em experiências químicas, sendo naturalmente aceite pelos investigadores dessa área de conhecimento.

Anotação

A anotação é um identificador literal utilizado pelo investigador na descrição de uma propriedade presente nos metadados que acompanham o conjunto de dados da sua investigação. Pode tomar qualquer valor, mas do ponto de vista semântico, e no âmbito desta tese, deverá indicar um conceito existente numa ontologia reconhecida pela comunidade. Por exemplo, a anotação "*nuclear magnetic resonance spectroscopy*" refere-se ao conceito com o mesmo nome na ontologia *Chemical Methods Ontology* (CHMO)[2].

Termo Semântico

O termo semântico é uma anotação que para além de uma identificação literal de uma propriedade, tem também indicada uma referência a um recurso externo, com auxílio do endereçamento único (URI) de um conceito, ou termo, presente em uma ontologia. Por

[2] http://www.ontobee.org/ontology/CHMO

exemplo, o URI "*http://purl.obolibrary.org/obo/CHMO_0000591*" referencia uma ligação que deve ser seguida por quem necessitar de compreender o significado do termo "*nuclear magnetic resonance spectroscopy*" através da interpretação dos factos de integração do conceito no seu domínio de conhecimento, expressados em RDF, apresentados através da navegação por HTTP ao URI. Esta ligação pode ser utilizada por qualquer investigador que pretenda classificar uma propriedade dos seus metadados com o mesmo conceito ontológico. Isto permite a remoção da ambiguidade da sua identificação, dado que cada conceito é totalmente independente do contexto em que é inserido, pois o seu significado é dado por um modelo formal externo. Através deste tipo de anotação é possível implementar o conceito de *Linked Data* e respetivos princípios já descritos na integração e partilha de dados através da *Web* Semântica.

Especificidade

As relações entre termos de uma ontologia apresentam, de modo geral, uma estrutura hierárquica (um grafo acíclico direto que permite uma ordem entre os termos). Isto possibilita que a ligação entre termos possa ser considerada como uma relação de classe-subclasse. Uma classe pode ter várias subclasses e a uma subclasse podem corresponder várias superclasses. Podemos assim considerar uma subclasse como uma especialização das suas classes ascendentes, o que no quadro de representação de um domínio de conhecimento e na escolha de um termo que melhor identifique uma propriedade de metadados representa um aumento na quantidade de informação conferida pela anotação e, portanto, uma evolução qualitativa de anotação de metadados.

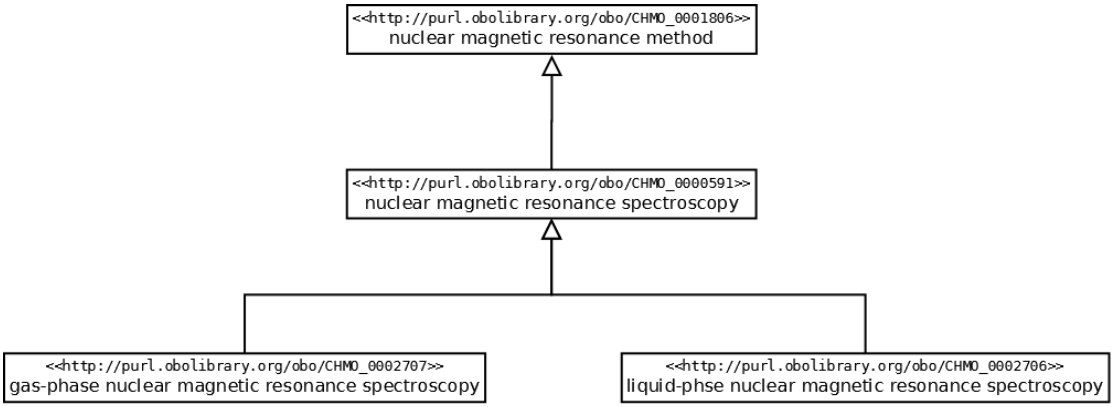


Figura 2-2 – Representação da estrutura hierarquica de uma ontologia.

Considerando a ontologia CHMO e a estrutura hierárquica particular do termo "*nuclear magnetic resonance spectroscopy*" como exemplo, representado no diagrama da Figura 2-2, podemos verificar que este tem pelo menos dois termos descendentes, ou subclasses: o "gas-phase nuclear magnetic resonance spectroscopy" e o "*liquid-phase nuclear magnetic resonance spectroscopy*"; e um ascendente: o "*nuclear magnetic resonance method*". Do ponto de vista da qualidade de integração semântica, a utilização

deste termo como anotação da descrição de uma propriedade de metadados poderia ser melhorada, na medida em que existem ainda subclasses que podem ajudar na especialização do conhecimento obtido, por parte de quem tenta integrar o conjunto de dados ao qual os metadados pertencem. No entanto, esse aumento de conhecimento pode não ser possível por não existirem termos mais específicos. A escolha de qualquer uma das subclasses como anotação resultaria num aumento do valor semântico dos metadados aos quais esta pertence, i.e., quanto maior for a especificidade da escolha do termo a utilizar, melhor será o processo de descoberta, compreensão e utilização dos metadados. A especificidade avalia assim a utilidade que uma anotação, ou termo, semântico, tem na descrição dos metadados (maior especificidade, maior utilidade), perante um investigador que pretenda integrar os dados, que estes metadados representam, em uma ação de investigação posterior.

Cobertura

Se as anotações, ou termos semânticos, podem ser utilizadas para relacionar propriedades de metadados e conceitos em ontologias externas, o número de anotações tem necessariamente de ser tido em conta para uma correta avaliação da utilidade, ou da qualidade de integração semântica, dos metadados. Para além da especificidade semântica das anotações, a contabilização da cobertura de anotações por termos semânticos é um fator de extrema importância na avaliação da integração semântica dos metadados. No melhor cenário, todas as anotações utilizadas na descrição das propriedades dos metadados estão cobertas por ligações a conceitos de ontologias externas; no pior cenário, nenhuma das anotações tem ligações com ontologias. No pior cenário, a utilidade dos metadados é tendencialmente nula, pois a compreensão das anotações utilizadas poderá ser potencialmente ambígua (p. ex., dois investigadores podem ter compreensões distintas sobre as descrições utilizadas). No melhor cenário, todas as anotações encontram-se descritas através de ligações a conceitos englobados em ontologias, que descrevem de modo não ambíguo um determinado domínio de conhecimento. Especificidade e cobertura formam assim os conceitos-chave para que a utilidade dos metadados (na sua descrição dos conjuntos de dados a quem os pretenda utilizar) possa ser avaliada, de modo a caraterizar aquilo que nesta tese se apelida de qualidade de integração semântica dos metadados.

2.3 Insuficiente caracterização semântica dos metadados

Para que a integração semântica de metadados seja um conceito totalmente aceite de modo a que a visão de Tim Berners-Lee de uma rede dados se torne num padrão de integração e partilha de informação científica, muitas barreiras existem ainda que devem ser ultrapassadas [14]. Seria de supor que estas barreiras fossem maioritariamente

tecnológicas, mas os esforços recentes na criação de repositórios de dados científicos, de ferramentas de criação de metadados e de anotação semântica, descritas na Secção 2.4, e de outras ferramentas de auxílio a ações de investigação permitem afirmar que a falta de utilização dos metadados sobretudo como ferramenta de descrição semântica terá outras razões.

O *Dryad*, p. ex., é um repositório de dados revistos manualmente, de suporte a publicações científicas e médicas. Teve início em 2008 e integra atualmente 468 jornais, 43864 autores de publicações e suporta 39122 conjuntos de dados respetivos [39]. É um repositório que tem como objetivo principal tornar os dados acessíveis à descoberta e à reutilização, mas sobretudo à citação. Os dados neste repositório recebem um identificador, o *Digital Object Identifier* (DOI), que à semelhança da função do URI identifica de modo único um determinado recurso. Este é geralmente utilizado na publicação de artigos de modo a que os seus leitores possam seguir o endereço e obter os dados da publicação. É focado sobretudo na citação de informação.

Apesar desta e de outras ferramentas tecnológicas, a maior barreira a transpor é de cariz social [3]. Agências de fundos públicos, instituições e jornais científicos têm vindo a tomar consciência da importância da integração e da partilha de informação para o avanço científico, o que se reflete nas politicas que impõem aos investigadores. Por exemplo, iniciativa *BioMedBridges*, um projeto comparticipado com fundos europeus, tem como objetivo a criação de pontes de dados e serviços a todas as *Biomedical Sciences Research Infrastructures* (BMSRIs) [40], através da definição de princípios de gestão e partilha de dados.

Mas a comunidade científica tem revelado pouco interesse e também dificuldade em participar nesse esforço. Poucos são os investigadores que consideram o tempo gasto na criação de descrições para os dados recolhidos [3], como produtivo, e isso reflete-se na qualidade geral dos metadados existentes em repositório. Esta falta de vontade surge, sobretudo, pelas atitudes e preocupações que os investigadores revelam sobre a integração e partilha dos seus próprios dados. E neste âmbito surge um conjunto de fatores que naturalmente influenciam o comportamento de cada um. Existem aqueles que revelam falta de destreza na utilização dos meios tecnológicos à sua disposição, p. ex., na identificação do repositório indicado (p. ex., o *Dryad*) e de que forma o utilizar. Existem aqueles que consideram simplesmente que a partilha de informação não é relevante para o seu trabalho. Existem aqueles que consideram a partilha relevante e necessária mas optam por não a fazer por ser demasiado dispendiosa do ponto de vista do trabalho envolvido e tempo despendido. Existem até aqueles que receiam que a partilha possa potenciar situações de abuso de utilização por parte de outros, sem o devido reconhecimento do mérito, ou receiam simplesmente uma outra qualquer nova situação.

Se a criação de metadados é um problema, a sua integração semântica torna-se ainda mais difícil. De acordo com um estudo realizado por um conjunto de alunos [16], da Faculdade de Ciências da Universidade de Lisboa (FCUL), que teve como caso de estudo o repositório de dados de investigação do EMBL-EBI, *MetaboLights* [15], a correta classificação semântica dos conjuntos de dados deste repositório está longe de ser a desejada e necessária. Tendo em conta os parâmetros que mediram, a (*i*) qualidade das anotações e a (*ii*) completude dos metadados, chegaram à conclusão que apenas 2/3 dos conjuntos de dados presentes no repositório estão semanticamente anotados, e que destes apenas 56% estão acima de uma fraca completude média de 64%. Apesar da boa qualidade das anotações, com 50% dos conjuntos acima de uma especificidade média de 84% na identificação dos termos, os autores concluíram ser necessário a tomada de medidas no sentido de garantir uma boa qualidade de anotações.

É pois necessário a criação de um mecanismo de reconhecimento e recompensa [41] daqueles investigadores que consideram como produtivo o tempo que passam na descrição do recursos gerados durante as ações de investigação, e mais ainda que o fazem de acordo com os princípios de *Web* Semântica. O objetivo da contribuição desta tese é desenvolver e implementar um algoritmo de avaliação da qualidade da integração semântica dos metadados e proporcionar a sua utilização através de uma plataforma, descrita na Secção 3.1 e na Secção 3.2.4, que acompanham esses recursos. Esta avaliação baseia-se na análise da especificidade semântica de cada uma das anotações utilizadas nos metadados, bem como na correta utilização de conceitos ontológicos.

2.4 Plataformas e ferramentas de anotação

Nesta secção é apresentada uma lista de plataformas e ferramentas que foram analisadas no âmbito da contribuição desta tese e que naturalmente se situam na integração semântica de estudos científicos. Foram consideradas duas ferramentas, ISA-TOOLS e ZOOMA, cujo objetivo se centra no auxílio aos investigadores na descrição dos seus dados de investigação, i.e., na criação, gestão e publicação dos metadados, acessíveis a outros investigadores sobre o paradigma da *Web* Semântica. Foi considerada uma plataforma, ISA-TAB, que estabelece a estrutura base para a construção de metadados complexos.

2.4.1 ISA-TAB

O formato *Investigation/Study/Assay* (ISA) tabular (TAB)[3] é um formato de utilização genérica para a recolha e comunicação de metadados de estrutura complexa

[3] http://isatab.sourceforge.net/format.html

(características de amostragem, tecnologias utilizadas, tipos de medição efetuadas) utilizados em experiências biomédicas, utilizando uma combinação de tecnologias, que estabelece à partida as condições para a integração semântica com vocabulários externos. Foi desenvolvido por uma equipa da Universidade de Oxford [42].

O conceito utilizado na especificação deste formato centra-se em três entidades distintas que dão corpo a uma ação de investigação científica: (*i*) investigação, (*ii*) estudo e (*iii*) ensaio. O seu principal objetivo é auxiliar o investigador na estruturação e classificação de toda a informação sobre o tema em estudo. Para tal, estabelece uma organização hierárquica daquilo que é considerado importante numa ação de investigação à qual corresponde um estudo, sobre o qual são elaborado vários ensaios. Esta estrutura lógica é transposta fisicamente ao modo como os dados de estudo são organizados na qual se encontra: (*i*) um ficheiro de investigação, (*ii*) um ficheiro de estudo e (*iii*) um ficheiro por ensaio.

O ficheiro de investigação é aquele que sumariza todo o estudo. Nele encontram-se definidas todas as entidades chave que compõem a investigação, através dele são relacionados os dados dos ensaios com o estudo em causa e é registada a ligação entre um ou mais estudos com uma investigação. Este ficheiro, pela informação que comtempla acaba por ser necessariamente os metadados de todos os estudos recolhidos através da plataforma ISA-TAB.

O ficheiro de estudo contem informação de contexto sobre os ensaios da investigação, p. ex., os assuntos estudados, a sua fonte, metodologia de amostragem e as suas características. Cada ficheiro de ensaio contém uma parte do fluxo de trabalho da investigação, assim como informação relacionada com a mesma.

Tabela 2-1 – Exemplo da estrutura metadados da especificação ISA-TAB

ONTOLOGY SOURCE REFERENCE	
Term Source Name	CTO
Term Source File	http://obo.sourceforge.net/cgibin/detail.cgi?cell
Term Source Version	
Term Source Description	The Cell Type Ontology
STUDY DESIGN DESCRIPTORS	
Study Design Type	time course design
Study Design Type Term Accession Number	OBI_11215
Study Design Type Term Source REF	OBI
STUDY PUBLICATIONS	
Study PubMed ID	17203948
Study Publication DOI	10.1021/pr0601640
Study Publication Author list	Griffin JL, Scott J, Nicholson JK.

Study Publication Title	The influence of pharmacogenetics on fatty liver disease in the wistar and kyoto rats: a combined transcriptomic and metabonomic study.
Study Publication Status	indexed for MEDLINE
Study Publication Status Term Accession Number	
Study Publication Status Term Source REF	
STUDY FACTORS	
Study Factor Name	Time
Study Factor Type	Temporal
Study Factor Type Term Accession Number	OBI_XXXx
Study Factor Type Term Source REF	OBI
STUDY ASSAYS	
Study Assay Measurement Type	Gene Expression
Study Assay Measurement Type Term Accession Number	OBI_XXXx
Study Assay Measurement Type Term Source REF	OBI
Study Assay Technology Type	DNA microarray
Study Assay Technology Type Term Accession Number	OBI_XXXx
Study Assay Technology Type Term Source REF	OBI
Study Assay Technology Platform	Affymetrix
Study Assay File Name	a_griffin123-Tx.txt
STUDY PROTOCOLS	
Study Protocol Name	standard procedure 1
Study Protocol Type	animal procedure
Study Protocol Type Term Accession Number	OBI_XXXx
Study Protocol Type Term Source REF	OBI
Study Protocol Description	All animal procedures conformed to Home Office, UK, guidelines for animal welfare. Male Wistar rats (n) 3 for each time point; control animals fed control diet …
Study Protocol URI	
Study Protocol Version	
Study Protocol Parameters Name	diet;population density
Study Protocol Parameters Term Accession Number	
Study Protocol Parameters Term Source REF	
Study Protocol Components Name	
Study Protocol Components Type	
Study Protocol Components Type Term Accession Number	
Study Protocol Components Type Term Source REF	

Da estrutura descrita no artigo [42] e sumarizada na Tabela 2-1, sobre o ficheiro de investigação importa realçar o esforço de integração da descrição do estudo com vocabulários externos, através da anotação de termos usados, com auxílio da chave *Term Accession Number*, no sentido de potenciar a sua compreensão e melhor utilização no âmbito da *Web* Semântica. Assim é de realçar as entidades: *Ontology Source Reference*,

onde são descritas as ontologias sobre as quais recaem as anotações de termos utilizados; *Study Design Descriptors* onde é indicado um termo e respetiva anotação para descrição do tipo de desenho implementado no estudo; *Study Publications* que informa sobre as publicações associadas ao estudo e sobre o estado de cada uma através de um termo anotado; *Study Factors* que enumera as variáveis utilizadas nos ensaios, as quais são categorizadas através de termos anotados; *Study Assays* que descreve os ensaios existentes na investigação e utiliza termos de vocabulários externos para a caraterização das medidas utilizadas e das tecnologias utilizadas na sua medição; *Protocols* que permite a utilização de anotações externas para a caraterização do tipo de protocolos utilizados na investigação, para a identificação dos parâmetros de cada protocolo e para a enumeração dos seus componentes.

Do ponto de vista da contribuição desta tese, este formato foi utilizado como conceito base para descrever aquilo que se considerou ser o resultado do motor de análise e avaliação, descrito na Secção 3.2.3.

2.4.2 ISA-TOOLS

A plataforma ISA-TOOLS[4] engloba um conjunto de ferramentas que possibilita a regularização da gestão local de metadados experimentais, permitindo a realização do método de revisão manual (*curation*) na fonte de metadados, suportando vários formatos normalizados de relatórios especificados pela comunidade, assim como a preparação dos estudos de modo a ser publicados em repositórios públicos.

Foi desenvolvida para tentar resolver dois problemas que surgiram com o aumento de requisitos de submissão de estudos feitos, por parte de patrocinadores e jornais científicos, na partilha de dados e na melhoria de conteúdo e normalização de metadados experimentais [43]: por um lado a diversidade de formatos de submissão, modelo de dados e terminologias existentes por cada área de investigação, que obrigava a que os investigadores formatassem os resultados em vários formatos, de modo a acomodar as exigências de publicação; por outro a falta de curadores que avaliassem e anotassem a submissão de resultados para os repositórios públicos. Tem como base o formato genérico de descrição, em formato tabular, de metadados ISA-TAB, desenvolvido em torno de três entidades chave: *Investigation*, *Study* e *Assay*.

O ISA-TOOLS permite aos seus utilizadores (p. ex., os investigadores) a compilação de metadados experimentais, com auxílio na procura e escolha de ontologias a utilizar e implementação de termos, assim como da sua integração com os dados experimentais. Para além de outras possibilidades, sobretudo relacionadas com a gestão do estudo, a

[4] http://www.isa-tools.org/software-suite

ferramenta possibilita a conversão direta para um conjunto de formatos próprios, utilizados por repositórios públicos.

Apesar do auxílio da escolha de termos a incorporar nos metadados experimentais, através da consulta em tempo-real de ontologias no *BioPortal* [44] (um dos maiores repositórios de ontologias na área da biomedicina), não é dado ao utilizador qualquer avaliação sobre a qualidade do termo que este pretende incorporar. O objetivo desta tese é precisamente classificar o valor semântico, ou qualidade de integração semântica, do termo perante outros que constituem o ramo de conhecimento que a ontologia oferece.

2.4.3 ZOOMA

A plataforma ZOOMA, desenvolvida pelo EMBL-EBI, permite a descoberta das melhores anotações para termos utilizados na descrição de conjuntos de dados experimentais, através da consulta de um repositório de anotações recolhidas durante o processo de integração semântica de um conjunto de bases de dados experimentais [45]. As anotações apresentadas foram todas elas manualmente revistas pela equipa interna. Pode ser utilizada como anotador automático de termos com elementos semânticos.

A fonte de anotações que a plataforma disponibilizada foi construída com base no esforço de anotação e revisão manual, i.e., de relacionamento manual entre termos e respetivas ontologias, de um conjunto de dados experimentais acerca de espécies, componentes anatómicos, tipos de células, tratamento de abuso de drogas e compostos, doenças e fenótipos, entre muitos outros.

A plataforma é assim um anotador de descrições utilizadas nos metadados, que procura a melhor correspondência, com um grau de confiança e qualidade que varia entre "alto" e "baixo", na sua fonte de dados de anotações revistas manualmente. Apesar da certeza que tenta oferecer sobre a melhor relação semântica dos termos utilizados, através da indicação de anotações para termos em ontologias públicas, esta não atribui nenhuma relevância à escolha do termo em si, i.e., não informa o utilizador da qualidade semântica do termo escolhido, de acordo com a sua especificidade, i.e., a sua posição na ontologia de origem.

À semelhança da plataforma ISA-TOOLS, a plataforma ZOOMA oferece meios de apoio à descrição de dados experimentais, neste caso através da sugestão de uma possível ontologia onde o termo se possa incluir, com tradução do seu URI, mas sem nunca ajudar o investigador na qualidade da informação que este pretende disponibilizar. A contribuição desta tese pretende informar o seu utilizador da qualidade semântica das anotações que este escolheu para descrever o seu trabalho. Se quisermos, após obter uma lista de anotações por parte do ZOOMA, o investigador pode usar a solução

implementada nesta tese para obter uma análise sobre a qualidade semântica de cada uma das anotações obtidas.

2.5 Ferramentas, plataformas e padrões utilizados

Ao longo desta secção são apresentadas as ferramentas, plataformas e padrões de desenvolvimento utilizados no desenho e implementação da contribuição desta tese. É feita uma introdução sobre a plataforma para de seguida indicar de que forma esta foi enquadrada no desenvolvimento da solução.

2.5.1 OWLtoSQL

O OWLtoSQL foi desenvolvido [46] como um programa Java que utiliza a biblioteca OWL-API [47] para carregamento e análise de ontologias em formato OWL 2 [37] de modo a extrair informação que se considere útil e salvaguardá-la em uma base de dados relacional. Lê informação de configuração de um ficheiro, em formato JSON, que contem os dados necessários de ligação ao motor de base de dados a utilizar como repositório, uma lista de ontologias a carregar em memória e uma lista de extratores de informação a utilizar durante o processamento.

Utiliza o conceito de extrator de modo a enumerar o tipo de informação a extrair das ontologias e salvaguardar na base de dados. Este é implementado através da realização e compilação de classes Java, que uma vez indicadas no ficheiro de configuração, são utilizadas de modo a navegar pela estrutura colocada em memória pela biblioteca OWL-API. O modelo relacional utilizado no OWLtoSQL, detalhado na Secção 3.3.1, centra-se na representação hierárquica de uma ontologia, através da análise das relações classe-subclasse dos termos que a compõem.

Uma das melhores capacidades desta ferramenta reside no facto de possibilitar que o acesso a uma ontologia, que pode ter um tamanho consideravelmente grande, possa ser feito de um modo mais eficiente, através de SQL, e de modo mais eficaz, apenas contenha aquilo que seja necessário. Esta aplicação é de grande importância na arquitetura da contribuição desta tese. Através desta ferramenta será possível a salvaguarda local de um conjunto de ontologias num modelo que pode ser rapidamente adaptado às necessidades da solução em causa, como é explicado na Secção 3.3.1.

O modelo de *In-Memory Database* (IMDB) foi considerado como alternativa ao modelo implementado pelo OWLtoSQL, de salvaguarda permanente da informação em disco. Apesar de apresentar uma melhor performance de resposta, o modelo IMDB não persiste a informação para um meio que possibilite falhas de execução do servidor. O tempo de conversão das ontologias, do formato OWL para um modelo relacional, é considerável. Com o modelo IMDB seria necessário converter o conjunto de ontologias,

a utilizar pela solução, para o modelo relacional em memória sempre que o motor que o suporta fosse reiniciado.

2.5.2 Desenvolvimento Java

Nesta secção são apresentados padrões de desenho e estilos arquiteturais de *software*, assim como métodos implementados ao longo do desenvolvimento da solução de contribuição desta tese.

Padrão Object-Oriented

O padrão *Object-Oriented Programming* (OOP) baseia-se no conceito de objetos para representação de entidades do mundo real, que podem conter dados, através da definição de propriedades que os caraterizam e métodos que indicam as ações a executar sobre os valores destas. Estes objetos são definidos através de classes, a partir da qual são instanciados e englobados. Classe e instância são assim dois dos principais conceitos deste padrão. Um objeto refere um tipo específico, ou instancia, de uma classe.

Este padrão é inerente à linguagem de programação utilizada, o Java, para o desenvolvimento da interface computacional e do motor de análise e avaliação.

RESTful Web Services

O *Representational State Transfer* (REST) é um estilo arquitetural de desenvolvimento aplicacional no âmbito da *Internet*, cujo principal conceito se baseia na abstração dos detalhes e constrangimentos de implementação de componentes de uma arquitetura de sistema, para se focar no papel que cada componente desempenha, nos constrangimentos da interação com outros componentes no domínio do sistema e na sua interpretação de elementos de dados [48]. Esta comunicação acontece através do protocolo HTTP, por indicação de um URI. Os *Web Services* são interfaces computacionais que encapsulam um conjunto de funcionalidades e de dados, disponibilizados através da *Internet*. Os *RESTful Web Services* são interfaces computacionais que incorporam o paradigma REST, permitindo o acesso à sua funcionalidade através de HTTP e utilizando tanto os métodos HTTP (*POST*, *GET*, *PUT*, *PATCH* e *DELETE*) e o esquema URI para identificação das ações disponíveis.

Esta foi a arquitetura escolhida para implementação da interface computacional da contribuição desta tese. A alternativa a este conceito de *Web Services* seria o conceito *Simple Object Access Protocol* (SOAP). Nesta arquitetura a comunicação é feita por mensagens em formato *Extensible Markup Language* (XML), que englobam pedidos através do protocolo SOAP a interfaces computacionais. No entanto, a complexidade de implementação desta arquitetura é superior à necessária pelo conceito REST, pois requer

ao cliente que implemente o protocolo SOAP do seu lado, o que aumenta não só o tempo de implementação, mas também a complexidade da solução.

Do ponto de vista da sua utilização na contribuição desta tese, a escolha deste estilo de arquitetura deveu-se, sobretudo, à consideração de que a interface computacional da solução oferece aos seus clientes os recursos (identificados de modo único através de URI e acedidos por chamadas HTTP, e respetivos métodos, a um único e consistente interface) do motor de análise e avaliação como um todo, naquela que é a sua missão de avaliação da qualidade de integração semântica de metadados, e não como serviços ou operações individuais de lógica aplicacional.

Java Sockets

A intercomunicação entre processos, ou *Inter-process Communication* (IPC), é um dos aspetos mais importantes no desenvolvimento de aplicações distribuídas, como é o caso da solução desenhada para a contribuição desta tese. Através da intercomunicação é possível dois processos, a executar em ambientes de produção distintos, trocar mensagens entre si. Este conceito foi utilizado na solução, implementada na contribuição da tese, para garantir a comunicação entre a interface computacional e o motor de análise e avaliação.

Tendo em vista a plataforma utilizada para o desenvolvimento da solução, baseada em Java, foram consideradas duas alternativas para implementação do conceito IPC: (*i*) *Java Remote Method Invocation* (RMI) ou (*ii*) *Java Sockets*. O RMI implementa um modelo simples e direto para computação distribuída, através de Java, mas a sua implementação é complexa e envolve a configuração de um conjunto de serviços paralelos para manter a sua funcionalidade. Os *Java Sockets* podem ser incluídos e implementados diretamente na solução, através da utilização de um conjunto de bibliotecas *Java* [49]. Estas permitem o controlo sobre os pontos de ligação, ou *Socket*, da comunicação entre dois processos. Cada *Socket* é identificado por um IP e uma porta TCP. Através destes pontos são enviadas mensagens por cada um dos processos.

2.5.3 Desenvolvimento Web

Para implementação da contribuição feita nesta esta tese foi necessário recorrer a um conjunto de tecnologias *Internet*, de modo a cumprir com o desenho da arquitetura. A seguir são apresentadas as linguagens, padrões de desenvolvimento e ferramentas escolhidas e implementadas para a construção da camada de apresentação, detalhada na Secção 3.2.1. Centram-se sobretudo em ferramentas gratuitas e soluções com uma grande aceitação por parte da comunidade de desenvolvimento de soluções para a *Internet*. Isto permitiu construir uma solução sem quaisquer custos e reduzir substancialmente possíveis curvas de aprendizagem das tecnologias utilizadas.

Hypertext Markup Language (HTML)

A *Hypertext Markup Language* (HTML) é uma linguagem de anotações cuja especificação formal é feita por parte do W3C, com forte aderência por parte dos maiores fabricantes de navegadores de *Internet* no mercado. Desenvolvida inicialmente, por Tim Berners-Lee (1990), para a anotação de documentos científicos [50], apresenta-se hoje como linguagem padrão, com maior importância e implementação, na criação de interfaces de utilizador na *Internet*. Baseia-se no paradigma de anotação de documentos, de modo a estabelecer uma estrutura e um contexto semântico sobre o mesmo.

A estrutura assenta sobre elementos de HTML (p. ex., cabeçalhos, parágrafos ou listas ordenadas) através dos quais é possível criar uma hierarquia bem estabelecida de um documento e publicá-lo na *Internet*. O *Document Object Model* (DOM) é uma representação em formato de árvore invertida dos elementos que constituem um documento anotado através do HTML. Encontra-se neste momento na versão 5, com a recomendação pelo W3C de 28 de Outubro de 2014. Foi a linguagem escolhida para o desenho da interface de utilizador, descrito na Secção 3.2.1, que foi colocado em um dos nós aplicacionais da estrutura PaaS de suporte da solução, descrita na Secção 2.5.4.

Biblioteca jQuery

A plataforma *jQuery* é uma biblioteca de *JavaScript*, uma linguagem de alto nível suportada pela maioria de navegadores de *Internet*, cujo objetivo se centra na abstração das dificuldades inerentes à programação de funcionalidades na interface cliente de uma aplicação *Web* [51]. Esta plataforma é geralmente associada a uma página HTML e permite que a navegação e manipulação do seu DOM possa ser rapidamente feita através da abstração dos problemas geralmente associados aos motores de navegação e respetivas versões. O *jQuery* permite a simplificação da procura, da seleção e da manipulação destes elementos, através de um conjunto de objetos que retiram ao programador o problema da sua implementação tendo em conta a especificidade dos múltiplos navegadores de *Internet* existentes.

Esta biblioteca foi utilizada para a construção das funcionalidades oferecidas pela interface de utilizador da contribuição desta tese. A interface de utilizador apresenta um paradigma de implementação assíncrono, i.e., permite que múltiplas ligações HTTP paralelas possam existir entre a interface e o servidor de suporte sem intervenção direta do utilizador. Este paradigma de desenvolvimento dá pelo nome de *Ajax* e caracteriza a utilização de um grupo de tecnologias na criação de aplicações no âmbito da *Internet*, com a capacidade de comunicação assíncrona com os servidores de suporte. Através da biblioteca *jQuery* é possível implementar uma camada de abstração sobre a dificuldade de implementação deste grupo de tecnologias, que compreende o *JavaScript*, o HTML e

o JSON/XML como linguagens e notações de implementação e o objeto *XMLHttpRequest* como interface cliente de comunicação assíncrona com o servidor remoto.

Padrão Model-View-Component

O *Model-View-Component* é um padrão de desenho, ou arquitetura, de interfaces de utilizador de computador [52]. Baseia-se na divisão de uma solução aplicacional em três camadas distintas, mas interrelacionadas: (*i*) o modelo de dados, (*ii*) a vista sobre a informação e (*iii*) o controlador de ações disponíveis; de modo a separar aquilo que é a representação da informação ao utilizador, da forma como ela é manipulada internamente. É um dos padrões de desenvolvimento mais utilizado no desenho de interfaces de utilizador para a *Internet* [53]. Por ser um padrão de desenho é independente da linguagem ou meio de implementação, sendo a sua utilização feita sobretudo em plataformas de desenvolvimento. Este é o padrão implementado pela plataforma de desenvolvimento, o *CakePHP*, escolhida para a construção da interface de utilizador da contribuição desta tese, descrita na secção seguinte.

Hypertext Preprocessor (PHP)

O *Hypertext Preprocessor* (PHP) é uma linguagem de *scripting* (método de escrita de programas autómatos para serem executados num determinado ambiente) desenvolvida especialmente para o desenvolvimento *Internet* com uma forte comunidade de utilizadores, que é interpretada por um servidor de conteúdos (p. ex., o Apache), através da implementação de um módulo ou através da utilização de um *Common Gateway Interface* (CGI), de modo a produzir uma resposta customizada para cada pedido de utilizador de um portal *Internet* [54].

Uma das melhores caraterísticas que possui baseia-se na possibilidade de ser embebida em código HTML, ou simplesmente utilizada de modo genérico para o desenvolvimento de gestores de conteúdo *Internet* ou plataformas de desenvolvimento para a *Internet*, como é o caso do *CakePHP*, detalhado nas secções seguintes. Foi a tecnologia utilizada para plataforma de construção da interface de utilizador.

Plataforma CakePHP

O *CakePHP* é uma plataforma de desenvolvimento para *Internet* [55], de acesso gratuito, mantida pela *Cake Software Foundation*. Foi desenvolvida na linguagem PHP, descrita no ponto anterior, e permite o rápido desenvolvimento e implementação de soluções com interface Web, como portais de *Internet* ou serviços Web. Com o seu desenvolvimento iniciado em 2005, apresenta-se hoje com uma forte comunidade de utilizadores. É uma plataforma que utiliza, e possibilita aos seus utilizadores o acesso a alguns conceitos e padrões de engenharia de *software* bem estabelecidos, sendo de realçar o padrão MVC,

descrito nos pontos anteriores, e o padrão *Convention Over Configuration*, cujo objetivo se centra na diminuição de decisões a ser tomadas por quem desenvolve a solução, aumentando a simplicidade e velocidade de uma solução. Esta foi a plataforma escolhida para a edificação da interface de utilizador de toda a solução.

Long Polling

O método de *Long Polling* é uma técnica desenvolvida para colmatar os problemas de comunicação derivados da arquitetura das aplicações em produção no ambiente *Internet*. Esta arquitetura baseia-se no modelo cliente-servidor, onde a comunicação parte sempre do cliente, que estabelece uma ligação com o servidor de modo a obter um resultado. Esta arquitetura impede que o servidor possa ter a iniciativa da comunicação, i.e., não possibilita que o servidor estabeleça uma ligação com cliente, sem que este a tenha requisitado, de modo a enviar-lhe informação.

O método *Long Polling* permite que o servidor mantenha uma requisição do cliente, por tempo indefinido, e a utilize para comunicar quando existe informação a ser enviada. O processo inicia-se com uma primeira requisição por parte do cliente. O servidor mantêm a requisição suspensa enquanto não existir nada para enviar ao cliente. Quando existirem dados para enviar, a requisição é utilizada para enviar uma resposta ao cliente. Este quando recebe a resposta envia uma nova requisição ao servidor e o ciclo repete-se.

Este método foi o utilizado para a comunicação entre a interface de utilizador e a interface computacional [56]. Foi utilizado de modo a permitir que as mensagens de operação do motor de análise e avaliação fossem entregues ao cliente.

2.5.4 Plataforma de suporte da solução

Esta tese foi desenvolvida no âmbito da *Web* Semântica e como tal um dos aspetos importantes a considerar foi a forma como a solução a desenhar, para a contribuição, poderia ser disponibilizada para demonstração e avaliação. Dado o volume de tecnologias que se previu utilizar na sua construção, foi necessária uma avaliação cuidada das hipóteses de suporte existentes. A solução encontrada tinha de corresponder a alguns critérios como: (*i*) o custo, (*ii*) a escalabilidade e (*iii*) a capacidade de resposta a um conjunto de necessidades tecnológicas. A escolha recaiu, naturalmente, sobre uma arquitetura *Cloud* de âmbito público [57], que permitisse o desenvolvimento e implementação das várias camadas que constituem a solução final num único ponto.

Dos três tipos genéricos de *Cloud Computing* [58]: *Infrastructure as a Service* (IaaS), *Plataform as a Service* (PaaS) e *Software as a Service* (SaaS), aquele que foi considerado ideal ao desenvolvimento, teste e colocação em produção da contribuição foi o PaaS. Este oferece uma plataforma com um conjunto de ferramentas e serviços desenhados de modo

a permitir uma rápida codificação e colocação em produção de soluções de *software*. Esta solução permite vários graus de liberdade, pois implementa um ambiente bastante personalizável de desenvolvimento, teste e implementação de aplicações. O IaaS apresenta apenas soluções para a administração de infraestrutura (i. e., servidores, redes de comunicação). O SaaS apresenta aplicações já em produção, desenvolvidas por terceiros, que podem facilmente ser geridas e utilizadas através da *Internet*.

Red Hat PaaS

A solução de *Cloud Computing* da *Red Hat*, o *OpenShift* [59], foi a escolhida. Esta implementa o paradigma PaaS [58], pois disponibiliza uma plataforma através da qual foi possível o desenvolvimento, o teste e a colocação em produção da solução, desenhada para contribuição da tese, num único ambiente através de um conjunto de ferramentas. Esta é uma implementação comercial, apesar de ter sido utilizada de modo gratuito, de um projeto comunitário, o *OpenShift Origin* de desenvolvimento de uma *Application Container Platform*, i.e., uma plataforma que implementa um padrão de separação lógica dos processos que nela se encontram a executar [60]. O conceito base é a *Application*, que engloba um conjunto de serviços escolhidos pelo cliente. Esta é suportada por uma infraestrutura por camadas especialmente desenhada para a partilha de recursos por várias aplicações, i.e., a execução em paralelo de um conjunto de serviços que partilham tudo aquilo que se encontra num único sistema operativo.

Na base da infraestrutura de suporte do *OpenShift* encontra-se o sistema operativo baseado no *Project Atomic* [61], um pequeno sistema operativo especializado na execução paralela de aplicações, com especial atenção na partilha de recursos existentes (memória, disco e processador). Este foi desenvolvido particularmente para suportar a execução da camada seguinte, na infraestrutura do *OpenShit*, a camada de segmentação aplicacional, na qual são executados os serviços dos clientes. Esta camada baseia-se na plataforma *Docker* [62] desenvolvida sob o conceito de contentores aplicacionais, áreas isoladas e totalmente independentes entre si, mas que partilham os recursos da camada abaixo, o sistema operativo.

Do ponto de vista das aplicações cada uma recebe um endereçamento público único (URL), constituído pelo seu nome particular e pelo nome dado ao conjunto de aplicações do cliente. O acesso dos clientes da aplicação é feita através de HTTP, pela porta TCP 80, sendo internamente redirecionada para a porta TCP 8080, a única disponível à aplicação para servir o seu conteúdo. No entanto, é possível aceder à aplicação através do protocolo SSH para efeitos de administração. Este protocolo permite ter uma janela sobre a consola do sistema operativo que suporta a execução dos serviços instalados. Para auxílio ao desenvolvimento todas as aplicações suportam a plataforma *Git* [63], para gestão de código fonte.

Toda esta infraestrutura oferece ao cliente um grau considerável de abstração sobre aquilo que teria necessariamente de ser feito, de modo a colocar uma solução em produção. Resta apenas ao cliente indicar as ferramentas que necessita para desenvolver ou implementar a solução que pretende, tudo sobre o conceito de *Application*. Para suportar a contribuição desta tese foram criadas duas aplicações na infraestrutura: (i) *masterweb*[5] e (ii) *masterrestfull*[6]. A primeira foi utilizada para o desenvolvimento e implementação da camada de apresentação, detalhada na Secção 3.2.1, na qual foi instalada um servidor de conteúdos, o *Apache* HTTP *Server*, descrito na secção seguinte, com suporte para PHP, linguagem referida na Secção 2.5.3. A segunda aplicação foi utilizada de três modos diferentes: (*i*) como suporte da camada *Web*, (*ii*) para execução do motor de análise e avaliação e (*iii*) suporte da camada de dados.

Para suporte da interface computacional, descrito na Secção 3.2.2, foi instalado um servidor aplicacional, o *Apache Tomcat Server*, descrito nas secções seguintes. Como suporte do motor de análise e avaliação, detalhado na Secção 3.2.3, apenas foi necessário utilizar o *Java Runtime Environment* (JRE) já instalado de raiz na aplicação. Para suporte do repositório de ontologias foi necessário utilizar um dos vários motores relacionais existentes. Para o efeito foi escolhido o *MySQL 5.5*, detalhado nas secções seguintes.

Servidor de conteúdos Apache

O *Apache* HTTP *Server* é um servidor de conteúdos através do protocolo HTTP, descrito na Secção 2.5.5, cujo desenvolvimento é mantido pela *Apache Software Foundation (ASF)*. Os conteúdos são agrupados em portais *Internet* e colocados para consulta através da *Internet* com auxílio de um navegador da *Internet* [64]. Estes conteúdos podem ser estáticos (p. ex., páginas de HTML com conteúdo fixo, imagens, documentos) ou dinâmicos (p. ex., páginas de HTML geradas em tempo real). Para gerar conteúdos dinâmicos é necessário incorporar módulos de *scripting*, capazes de interpretar linguagens de programação como o PHP, detalhada na Secção 2.5.3, de modo a construir a resposta a dar ao cliente.

Este servidor foi escolhido para suporte da interface de utilizador da contribuição desta tese pela facilidade de utilização, pelo desempenho e robustez que oferece na sua execução. Este pode ser consultado através do endereço *http://masterweb-metadataanalyser.rhcloud.com*. Para além disso é um dos servidores mais utilizados na *Internet* no suporte a portais e serviço de conteúdos, com uma quota de utilizadores perto de 50% [65].

[5] http://masterweb-metadataanalyser.rhcloud.com

[6] http://masterrestfull-metadataanalyser.rhcloud.com

Servidor aplicacional Tomcat

O *Apache Tomcat Server* é um servidor aplicacional através do protocolo HTTP, descrito na Secção 2.5.5, cujo desenvolvimento é mantido pela *Apache Software Foundation* (ASF) [66]. Um servidor aplicacional é geralmente responsável por aplicar a lógica de negócio numa solução multicamada, situado entre a interface ao utilizador e o repositório de dados. Esta lógica é desenvolvida com auxílio da linguagem Java e assume vários tipos aplicacionais. Os mais comuns são (*i*) os *Java Servlets*, módulos aplicacionais desenvolvidos através do paradigma *Object-Oriented*, cujo objetivo é servir os clientes de conteúdo dinâmico, i.e., conteúdo gerado por pedido; (*ii*) as *Java Server Pages* (JSP), utilizadas na criação de portais *Internet* dinâmicos, que servem de abstração à utilização dos *Java Servlets*, na criação de conteúdo para a *Internet*.

Este servidor foi utilizado para suportar a aplicação de interface computacional, descrito na Secção 3.2.2. Este pode ser consultado através do endereço *http:// masterrestfull-metadataanalyser.rhcloud.com*. Esta aplicação foi desenvolvida como um projeto *Java*, de âmbito *Web*, com recurso ao conceito de *RESTFull Web Services*, descrito na Secção 2.5.2. Uma vez compilado, foi colocado neste servidor de modo a que possa responder a pedidos externos, feitos através de HTTP. No entanto, para que possa ser executada como uma aplicação, ou como um *Servlet*, foi necessário implementar a plataforma *Jersey* [67]. O objetivo desta é estender a plataforma JAX-RS [68] (a biblioteca por defeito de implementação de *RESTFull Web Services* no Java) de modo a tornar mais fácil a construção deste tipo de solução adicionando, para tal, um conjunto de funcionalidades que a última não possui. Uma destas funcionalidades será utilizada no procedimento de *Long Polling*, cuja implementação se encontra descrita na Secção 3.2.2, na construção da interface computacional.

Esta arquitetura *Tomcat/Java* poderia ter sido facilmente substituída pela arquitetura *Apache/PHP*, para implementação das funcionalidades principais da interface computacional. Através da linguagem PHP é também possível desenvolver a lógica necessária para implementar o conceito de *RESTFul Web Services*, tendo como base um servidor HTTP como o *Apache*, descrito no ponto anterior. No entanto, a funcionalidade de *Long Polling* é implementada de modo mais simples e direto com auxílio das bibliotecas *Jersey* e *JAX-RS*, através da arquitetura *Tomcat/Java*, pois as ferramentas necessárias à sua utilização encontram-se já desenvolvidas em ambas, constituindo por isso um ganho de tempo e simplicidade na conceção da contribuição desta tese.

Base de dados MySQL

O *MySQL* é um motor de base de dados, que implementa o modelo relacional de salvaguarda de informação. O modelo relacional define uma base de dados como uma

coleção de uma ou mais relações, onde cada relação é uma tabela com linhas e colunas [69]. Este modelo de representação simples da informação permite uma rápida compreensão e utilização, através de linguagens de alto nível de pesquisa de dados, como o *Structured Query Language* (SQL), que se tornou especificação formal pela *International Organization for Standardization (ISO)* [70].

Com auxílio do SQL é possível interrogar o modelo de dados e obter a informação nele contido. Para além da capacidade de interrogação, esta linguagem dispõe ainda de dois outros grupos de ações sobre o modelo relacional: (*i*) *Data Definition Language* (DDL) e o (*ii*) *Data Manipulation Language* (DML). O primeiro grupo destina-se à criação e gestão do modelo, pois através dele é possível definir toda a estrutura, tabelas, propriedades e relações, necessárias ao modelo. O segundo destina-se à manipulação da informação contida no modelo. Através desta é possível a inserção, modificação e eliminação de informação nele contido.

Este foi o motor escolhido para a salvaguarda da conversão das ontologias, do seu formato original em OWL para um modelo relacional, descrita na Secção 3.2.4. Esta escolha deveu-se sobretudo pela implementação que a ferramenta de conversão, o *OWLtoSQL*, obrigou, i.e., o motor relacional alvo utilizado pela ferramenta é o *MySQL*.

2.5.5 Protocolos e Notações

Nesta secção são enumerados os protocolos que serão utilizados na comunicação dos clientes com a solução a desenvolver. Para além disso serão também descritas as notações a utilizar no resultado a enviar aos clientes.

Hypertext Transfer Protocol

O *Hypertext Transfer Protocol* (HTTP) é um protocolo de comunicação entre sistemas de informação distribuídos e colaborativos [71], ou entre um cliente e um sistema de informação. Desenvolvido pela iniciativa *World Wide Web* [72], faz parte do grupo de protocolos da *Internet*, situado na sua camada superior, o *Application Layer*. Como o próprio nome indica, é um protocolo desenvolvido para a transferência de dados, através da *Internet*, de hipertexto, vulgarmente apelidado de páginas *Internet*, i.e., é o protocolo de suporte à linguagem HTML, descrita na Secção 2.5.3. Da sua longa especificação existem dois aspetos que importa referir: (*i*) é um protoloco que não mantém sessão e (*ii*) define um conjunto de ações ou verbos, que pode ser indicado a cada chamada. Ambos terão implicações no desenho da solução da tese.

Não manter sessão significa que na comunicação entre duas entidades (cliente e servidor) através de HTTP não é retido qualquer tipo de informação pela entidade que

serve a informação, i.e., através deste protocolo o servidor não mantem qualquer tipo de informação sobre os seus clientes.

A lista de ações ou verbos permite estabelecer o critério de ação e resposta dada pelo servidor, ao pedido do cliente. Da lista de verbos, que pode ser consultada em [71], os mais utilizados são (*i*) o *GET* e o (*ii*) *POST*. O primeiro requer que seja enviada ao cliente uma representação do recurso identificado através do seu endereço, o URI. O segundo requer que a entidade associada ao pedido seja incluída na lista de subordinados no recurso identificado pelo endereço.

Quer a interface de utilizador, quer a interface computacional têm necessariamente de comunicar os seus estados através deste protocolo, uma vez que o protocolo não mantém o estado. Esta comunicação é feita perante pedidos GET, os quais serão respondidos com uma representação do último estado de cada um dos interfaces. No entanto, é necessário enviar um conjunto de valores à interface computacional, de modo a que este possa elaborar a análise e avaliação dos valores de especificidade dos metadados. O verbo aplicado para esta comunicação foi o POST. Esta operação POST e a falta de manutenção do estado de sessão terão de ser suportadas pela implementação do conceito *RESTFul Web Services*, detalhada na Secção 2.5.2

Comma Separated Values

O *Comma Separated Values* (CSV) é um formato de salvaguarda e troca de informação em ficheiro de texto, onde cada linha é um registo e cada registo apresenta um conjunto de campos. O carater utilizado para a separação destes campos é a vírgula. Apesar de não existir uma especificação oficial deste ficheiro existe um *Request For Comments* (RFC), por parte da *The Internet Engineering Task Force* (IETF), que é normalmente utilizado como referência [73].

JavaScript Object Notation

O *JavaScript Object Notation* (JSON) é um formato público de troca de informação que incorpora objetos de dados numa estrutura de pares atributo-valor, através de listas ordenadas [74]. Tem um duplo objetivo: a facilidade de compreensão por humanos e por meios computacionais. Apesar de ser suportado por muitas linguagens de programação, o seu formato apresenta-se na última especificação ECMA como independente de qualquer linguagem. No entanto, foi originalmente baseado no *JavaScript*.

Foi a notação escolhida para a implementação do protocolo de comunicação com a interface computacional, da solução construída como contribuição desta tese. A escolha deveu-se sobretudo pela adoção desta notação como padrão na comunicação entre aplicações na *Internet*.

2.6 Sumário

Nesta secção foi apresentado o trabalho relacionado com o desenvolvimento desta tese, tendo sido descritos os conceitos base sobre a *Web* Semântica, e iniciativas posteriores, assim como um conjunto de ferramentas que se englobam no contexto do problema que esta tese apresenta. Foi também apresentado o conceito de integração semântica de metadados e enumeração dos elementos necessários à sua concretização.

No capítulo seguinte será apresentado o trabalho desenvolvido no âmbito desta tese. Este baseia-se no estudo sobre medidas de avaliação da qualidade de integração semântica dos metadados de um conjunto de dados, e na implementação de uma plataforma que as implementa, proporcionado ao seu utilizador, humano ou computacional, um meio de avaliação geral da qualidade de integração semântica dos metadados que acompanham conjunto de dados.

Capítulo 3

Trabalho desenvolvido

Neste capítulo é apresentado o trabalho desenvolvido para a contribuição desta tese, tendo em conta o problema enunciado na introdução sobre a avaliação semântica de metadados, que envolveu a edificação de uma plataforma que permite a avaliação da qualidade de integração semântica de metadados, de acordo com os conceitos de especificidade e cobertura descritos na Secção 2.2, com posterior emissão de relatório do processo de análise e avaliação, de modo global, com um conjunto de vocabulários públicos ou ontologias, tendo em conta (*i*) a especificidade das anotações que apresentam uma referência a conceitos ontológicos (termos semânticos) e (*ii*) a razão entre termos semânticos e restantes anotações.

Para tal, foi necessário definir formalmente as medidas de especificidade de um termo semântico tendo em conta a posição do conceito ontológico indicado, na sua ontologia de referência, e de cobertura de termos semânticos tendo em conta o número de anotações que fazem referência a conceitos ontológicos e todas as outras anotações, utilizadas na descrição dos conjuntos de dados. Este estudo destas medidas é apresentado na Secção 3.1, a implementação da plataforma é apresentada na Secção 3.2.

3.1 Medidas de avaliação da qualidade dos metadados

A avaliação da qualidade de integração semântica dos metadados proposta por esta tese tem em conta duas medidas: (*i*) o nível de especificidade dos termos semânticos utilizados para anotação dos conjuntos de dados, do ponto de vista do seu enquadramento nas ontologias a que pertencem; (*ii*) o grau de cobertura (abrangência) do número de anotações com referência a conceitos ontológicos (termos semânticos), perante o número total de anotações utilizadas na descrição dos metadados.

Esta avaliação parte do conceito de ontologia como modelo relacional, no qual é representada como um grafo direto acíclico, ilustrado na Figura 3-1, onde é possível descrever a relação entre ascendentes e descendentes de um determinado termo da ontologia, como uma árvore invertida e considerar a sua especificidade como função da sua a posição relativa num determinado ramo da mesma, i.e., o termo encontra-se posicionado num dos ramos da árvore, que no melhor dos casos será apenas um, como ilustrado na Figura 3-1 (a). Neste caso, a especificidade de um conceito é determinado pela profundidade do termo no ramo (como a razão entre a distância do termo até à raiz

da árvore, e a distância do último termo, ou termo folha, no ramo até este). Para uma árvore com mais do que um ramo, como ilustrado na Figura 3-1 (b), é necessário ter em conta o cumprimento de todos os ramos no qual o termo participa, i.e., é necessário considerar a distância média entre si e todos os termos folha que possam existir em cada

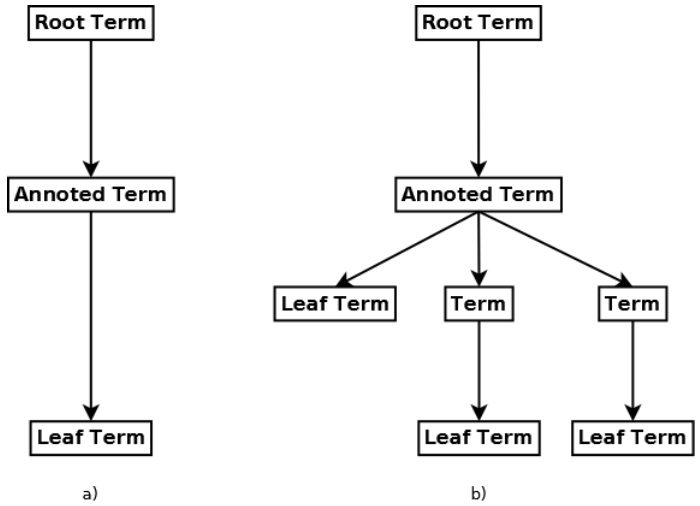


Figura 3-1 - Representação de uma ontologia através de um grafo aciciclo.

ramificação construída através de todos os seus descendentes.

Assim, seja $T = \{t_1, t_2, \ldots, t_m\}$ o conjunto de todos o termos da ontologia, presentes em um ficheiro de metadados. Para cada $t \in T$, o valor de especificidade da anotação desse termo $S(t)$ é calculado utilizando a seguinte equação:

$$S(t) = \frac{A_t}{A_t + D_t}$$

Onde At é o número de termos ascendentes de t e Dt a distância média entre t e todos os seus descendentes folha, $L = \{l1, l2, \ldots, lz\}$, i.e., todos os descendentes que não tenham qualquer relação de subclasse com outros termos, calculada através da equação:

$$D_t = \frac{1}{z} \cdot \sum_{i=1}^{z} b_i$$

Onde z é o número de termos descendentes de t que são termos folha, e bi é a distância entre t e a folha i. Termos que não pertençam a uma certa ontologia recebem um valor arbitrário de especificidade de -1.

O valor da função entre 0 e 1 indica um intervalo de relacionamento do termo na árvore. No seu limite inferior o termo está posicionado no topo da árvore, ou raiz, como demonstrado na Figura 3-1, o que denota que este não tem quaisquer ascendentes, $A_t = 0$. Este valor traduz uma má anotação pois existem outros termos mais específicos e por isso mais esclarecedores do significado da anotação. Uma melhor abordagem seria a utilização

de um termo ontológico mais específico, do ponto de vista da sua posição no ramo da árvore, fornecendo assim um valor semântico mais rico. No seu limite superior o termo está posicionado no fundo da árvore, perto de termos folha, o que denota que o termo não tem descendentes, $D_t = 0$. Este valor traduz o melhor dos resultados pois não é possível aprofundar a especificidade ou significado semântico que se pretende atribuir a uma anotação de metadados.

De modo a determinar a média de especificidade e grau de cobertura dos metadados, de um conjunto de dados, apenas as anotações para as quais o valor de especificidade, S_t, foi encontrado são consideradas, i.e., $S(t) \neq -1$. Deste modo, seja $A = \{a_1, a_2, ..., a_m\}$, o conjunto de todas as anotações encontradas em um ficheiro de metadados D, de modo a que $S(a_i) \neq -1 \; \forall i$. A média de especificidade de D, A_{avg}, é dada pela seguinte equação:

$$A_{avg} = \frac{1}{m} . \sum_{i=1}^{m} S(a_i)$$

O grau de cobertura é determinado pela razão entre o número de anotações presentes para as quais foi possível determinar um valor de especificidade, $S(t) \neq -1$, e o número total de anotações utilizadas na descrição dos metadados, D. De modo a calcular o grau de cobertura T, seja A o número total de anotações utilizadas em D e B o número de anotações com referência a conceitos ontológicos, o valor de T é dado pela seguinte equação:

$$T = \frac{B}{A}$$

Ambas as equações medem a qualidade de integração semântica dos metadados associados a um conjunto de dados: por um lado, como foi apresentado, quanto mais alto for o valor de especificidade de uma anotação, melhor é a descrição dos dados à qual ela se refere, assim como será a média de especificidade de todas as anotações presentes nos metadados; por outro lado, quanto maior for o grau de cobertura, de todas as anotações existentes por anotações com referência a conceitos ontológicos, maior será o conhecimento com significado computacionalmente tratável oferecido pelos metadados a quem pretende utilizar os conjunto de dados que eles descrevem.

3.2 Plataforma de análise e avaliação de metadados

Nesta secção é apresentada a arquitetura de implementação e funcionalidades inerentes da plataforma de análise e avaliação de metadados, estabelecida como contribuição desta tese. Sendo a medição da qualidade de integração semântica de metadados o principal requisito desta plataforma, foi necessário definir uma arquitetura que permitisse não apenas concluir o propósito de análise e avaliação do conteúdo de um ficheiro de metadados, mas também englobar as fontes de dados a partir da qual a avaliação é feita, assim como suportar uma estrutura que permitisse a interação com essa componente.

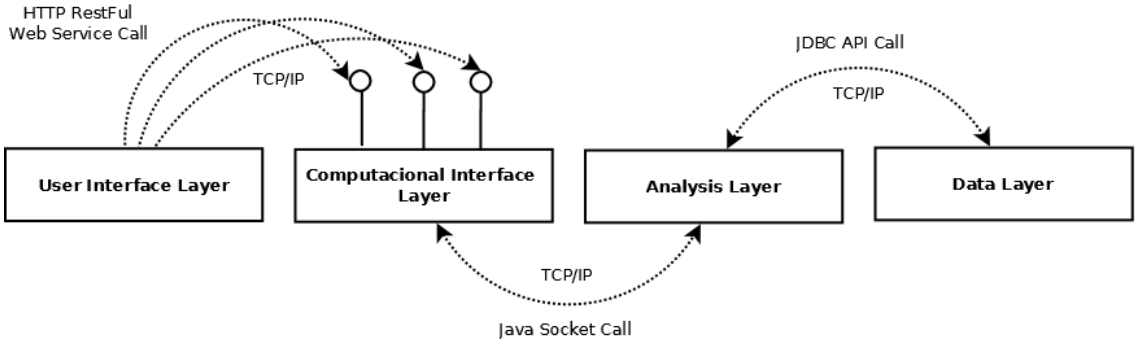


Figura 3-2 - Vista de camadas sobre a solução de contribuição da tese.

Com este propósito, foi desenhada uma arquitetura distribuída, representada na vista de camadas da Figura 3-2, expansível e com alta coesão, composta por várias camadas, que embora independentes colaboram entre si de modo a tornar possível a submissão de um objeto para avaliação e posterior recuperação dos resultados de análise e avaliação. Foram identificados as seguintes camadas de arquitetura:

- **Camada de Apresentação**: Apresenta uma interface ao utilizador da solução, de modo a permitir a submissão de ficheiros e obtenção de resultados de forma mais *user-friendly* possível. Esta interage com a camada *Web* através de chamadas HTTP, definidas sobre a nomenclatura *RESTFul Web Services* (Secção 3.2.1).

- **Camada *Web***: Define uma abstração de funcionamento direto com o motor de análise e avaliação, através da oferta de um conjunto de ações que podem ser invocadas computacionalmente (Secção 3.2.2). Interage com a camada de análise através de chamadas *Java Socket* sobre o protocolo TCP/IP.

- **Camada de Análise**: Define a composição do motor de análise e avaliação, que recebe o ficheiro de metadados e efetua sobre ele um conjunto de ações de modo a efetuar uma avaliação do ponto de vista da sua especificidade e

cobertura (Secção 3.2.3). Interage com a camada de dados através da biblioteca JDBC.

- **Camada de Dados**: Apresenta o repositório de informação sobre as ontologias que são suportadas, num dado momento, pela solução (Secção 3.2.4).

3.2.1 Camada de Apresentação

A camada de apresentação foi definida de modo a englobar os interfaces disponíveis à interação entre um utilizador humano e os serviços que a plataforma pretende oferecer. Esta camada tem como principal objetivo a abstração do utilizador do sistema da complexidade de análise e avaliação de um ficheiro de metadados por ele submetido. Nesse sentido foi elaborado uma única interface em linguagem HTML baseado na plataforma de desenvolvimento rápido *CakePHP*, ambas descritos na Secção 2.5.3. A interface encontra-se hospedada na plataforma *Cloud* da *Red Hat*, descrita na Secção 2.5.4, e publicado com um endereço público: *masterweb-metadataanalyser.rhcloud.com*.

A plataforma *CakePHP* apresenta um padrão de desenho MVC, descrito na Secção 2.5.3, e foi utilizada para a definição de uma única página inicial. Esta encontra-se dividida horizontalmente por cinco secções: (*i*) introdução, (*ii*) formulário de envio de ficheiros (ilustrado na Figura 3-3), (*iii*) características da interface, (*iv*) acerca da solução e (*v*) contactos. O formulário de envio de ficheiros, a secção principal, é composto por quatro elementos: (*i*) o repositório de metadados ao qual o ficheiro a ser submetido pertence, (*ii*) o ficheiro físico a ser enviado à plataforma para avaliação, (*iii*) o endereço público de um ficheiro de metadados e (*iv*) o botão de envio do formulário à plataforma.

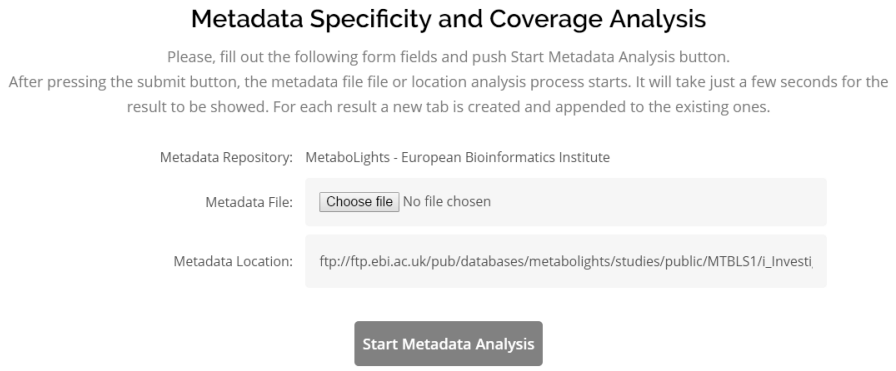


Figura 3-3 - Formulário de envio de ficheiros do interface de utilizador

Após envio do formulário, e até à entrega final de resultados, é apresentada uma área de progresso do processo de análise e avaliação (ilustrado na Figura 3-4) do ficheiro indicado. Esta enumera as fases principais de processamento, que se inicia com uma mensagem de reconhecimento de entrega com sucesso do formulário e se estende até à conclusão do processo de cálculo, detalhado na Secção 3.2.3. Após a última mensagem é construída na interface uma secção onde é representado, em HTML, o resultado da medição da especificidade e cobertura de todos os elementos encontrados no ficheiro de metadados. Esta representação tem como base o protocolo de resultados dos métodos da interface *REST*, detalhado na Secção 2.5.3. Para cada resultado é dada ainda a hipótese de exportação do mesmo, nos formatos CSV e JSON, descritos na Secção 2.5.5. Múltiplos resultados podem ser obtidos em acumulação, sendo que os novos resultados são adicionados à zona inferior da área de resultados.

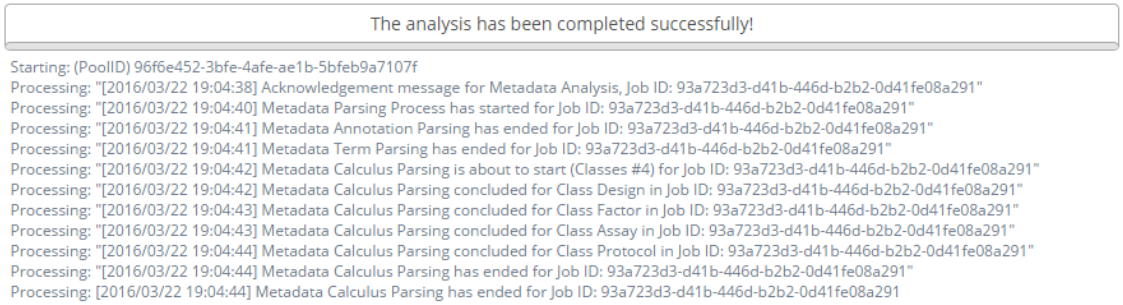


Figura 3-4 - Descrição de etapas de processamento na área de progresso.

A interação entre esta camada e a camada subjacente na arquitetura, apresentada na Figura 3-2, é feita por chamadas HTTP sobre ligações TCP/IP. Por cada chamada é indicado o ponto de contacto e a ação que se pretende ver realizada.

3.2.2 Camada *Web*

A camada *Web* foi definida de modo a apresentar um interface estritamente computacional, que expõe o estado e a funcionalidade através de um conjunto de recursos do motor de análise e avaliação, que os clientes podem manipular. A decisão de construção desta camada deveu-se sobretudo à necessidade de integração da operacionalidade do motor num ambiente mais alargado. Este interface foi desenvolvido através da implementação do paradigma REST sobre um componente de *Web Services* desenvolvido em *Java*, descrito na Secção 2.5.2.

Tabela 3-1 - Lista de ações disponíveis na interface computacinal.

Método	Verbo HTTP	Tipo de Dados de Retorno
/tojson/{poolid}	GET	JSON
/version	GET	JSON
/polling/{poolid}	GET	JSON
/submitfile	POST	JSON

Através do contato a um ponto único (*endpoint*), os clientes invocam ações e obtêm os resultados das mesmas. Este contacto é feito com auxílio do protocolo HTTP através da indicação de um URI. Esta camada apresenta quatro métodos, ou ações, descritos através de endereços únicos que são adicionados ao URI do serviço, enumerados na Tabela 3-1 e ilustrada pelo diagrama de casos de uso na Figura 3-5, p. ex., *http://masterrestfull-metadataanalyser.rhcloud.com/app/metadata/version*. Este interface encontra-se hospedado na mesma plataforma da camada de apresentação.

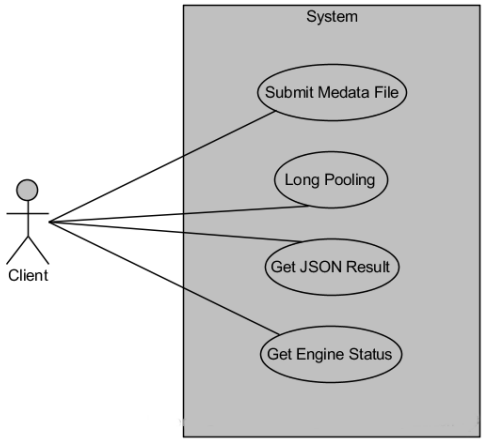


Figura 3-5 - Diagrama de casos de uso do interface computacional.

Cada método define assim um ponto de identificação único, na *Internet*, que o cliente, computacional ou humano, pode invocar e indicar a ação que pretende ver executada pelo serviço. O resultado é apresentado em notação JSON, descrita na Secção 2.5.5, que permite a junção da informação com a descrição da sua estrutura. Esta notação foi escolhida, em detrimento do XML, por apresentar uma maior facilidade de integração com outras linguagens de programação. Cada um dos métodos estabelece a seguinte função, ilustrada pelo diagrama de casos de uso na Figura 3-5:

- **Método *<u>/submitfile</u>***: Este método permite que um ficheiro seja submetido para análise e avaliação através do verbo POST do protocolo HTTP, descrito na 2.5.5. O ficheiro pode ser indicado na íntegra, em modo binário, ou através do seu endereço público na *Internet* (URL). Como retorno, o cliente obtêm uma mensagem em formato JSON, cuja representação se encontra descrita

na Secção 2.5.5, indicando (*i*) o sucesso do início do processo de análise e medição e o ID que serve para identificar, no servidor, esta tarefa de medição, que é usado para pedir informação acerca do grau de completude da tarefa (a primeira mensagem da área de progresso, ilustrada na Figura 3-4), ou (*ii*) a ocorrência de uma exceção que impossibilitou que o processo se iniciasse.

- **Método *`/polling/{poolid}`***: Este método permite a implementação da técnica de *Long Polling*, descrita na 2.5.3, possibilitando que as mensagens oriundas do motor de análise e avaliação cheguem até ao cliente, possibilitando assim a operação da área de informação ao cliente através da receção de múltiplas mensagens. É inicialmente chamado no fim do processo de envio do formulário do cliente e possibilita que seja mantido um ciclo à condição através da sua chamada recursiva. A condição é mantida através do valor do campo *type* do protocolo de resultados descrito na Secção 3.2.3.3. Este deverá conter um valor que indique o final do processo. Recebe como parâmetro o *PoolID*, um valor único gerado pela primeira chamada ao método *submitfile* e que serve como identificador da secção de trabalho entre cliente e a camada *Web* da plataforma.

- **Método *`/tojson/{poolid}`***: Este método permite que seja feita a exportação de resultados de análise para o cliente. Esta é feita através do formato JSON, detalhado na Secção 2.5.5, e representa o resultado da análise e medição de um ficheiro submetido anteriormente através do método *submitfile*. Recebe como parâmetro de entrada o valor *PoolID* e que identifica de modo único o resultado. Este valor é gerado pelo motor de análise e avaliação.

- **Método *`/version`***: Este é método de controlo e que permite a obtenção de um conjunto de informações acerca do estado do *Web Service* de suporte.

A interação entre esta camada e a camada de análise é feita através do conceito IPC de *Java Sockets*, detalhado na 2.5.2 e ilustrada na Figura 3-2. Através da indicação de um endereço IP e de uma porta TCP, uma ligação TCP/IP é estabelecida e enviada uma mensagem para o componente do motor de análise e avaliação. Esta é, no entanto, uma comunicação bidirecional dado que este componente pode estabelecer e receber ligações com os componentes da camada imediatamente abaixo.

3.2.3 Camada de Análise

A camada de análise foi estruturada de modo a implementar o motor de análise e avaliação (MAA), que constitui o principal elemento de contribuição desta tese. Cabe-lhe a receção de um ficheiro de metadados, a pesquisa sequencial de cada uma das linhas presentes e a

identificação e extração de todos os elementos necessários ao processo de avaliação e cálculo das medidas identificadas na Secção 3.1: (*i*) a especificidade da anotação e (*ii*) a cobertura de termos anotados. De modo a cumprir com a sua função foram implementados dois grandes procedimentos que o motor tem de efetuar por cada ficheiro de metadados recebido:

- **Procedimento de Análise (PAN):** Procedimento para identificação de todas as anotações utilizadas na composição de um ficheiro de metadados.

- **Procedimento de Avaliação (PAV):** Procedimento para de avaliação da especificidade de todos os termos semânticos encontrados, com posterior cálculo de médias globais de especificidade e cobertura de termos semânticos.

A especificidade é um dos fatores mais importantes de toda a estrutura, apenas determinada pela consulta de um repositório de ontologias. É calculada tendo em conta a posição do termo no grafo utilizado para representar cada um dos termos da ontologia correspondente e respetivas relações. Para tal, são contabilizados todos os antecessores do termo, assim como todos os seus descendentes. A razão entre estes dois elementos é normalizada de modo a que se situe entre 0 e 1. O valor 0 aponta para um termo muito genérico (baixa especificidade), pois indica que a sua posição é de topo, ou raiz, na árvore que constitui o grafo de relações entre um determinado conjunto de termos na ontologia. O valor 1, pelo contrário, aponta para um termo bastante específico, sobre o mesmo critério do valor anterior, pois indica que a posição do termo é de base, ou folha, na árvore descrita.

Tabela 3-2 - Lista de requisitos funcionais da solução de análise e avaliação.

Índice	Descrição
RF01	**Receção um ficheiro de metadados**, com identificação explícita do repositório de metadados a que pertence e efetuar a avaliação da especificidade de cada um dos termos encontrados, assim como a avaliação da cobertura de termos anotados perante aqueles que não o são

Apostando na simplicidade de utilização do motor apenas foi identificado um único requisito funcional, descrito na Tabela 3-2, e três requisitos não-funcionais ou de qualidade, descritos na Tabela 3-3.

Tabela 3-3 - Lista de requisitos não-funcionais da solução de análise e avaliação.

Índice	Descrição
RNF01	**Performance** – O sistema deverá responder a um pedido de análise e avaliação, em operação normal, com uma latência média inferior a 60 segundos
RNF02	**Usabilidade** – O utilizador deverá conseguir submeter um ficheiro de metadados, em modo operacional, e utilizar o seu resultado num intervalo de 120 segundos de utilização da aplicação
RNF03	**Modificabilidade** – Em modo de desenho, uma alteração aos interfaces de repositório deverá ser feita, testada e implementada num tempo máximo de 3 horas.

O motor foi desenvolvido através da estrutura tecnológica que o *Java* proporciona, em particular pelas suas características *Object-Oriented*, descrito na Secção 2.5.2. O padrão de desenho utilizado para o seu desenvolvimento foi o padrão *Blackboard*. Este padrão tem como característica principal a definição de um repositório global de variáveis, ou mensagens, que podem ser lidas e escritas por processos autónomos de processamento. Cada um destes processos participa assim na resolução de um problema comum a todos, bastando para isso consultar o repositório central de mensagens e identificar aquelas que de algum modo lhe interessam.

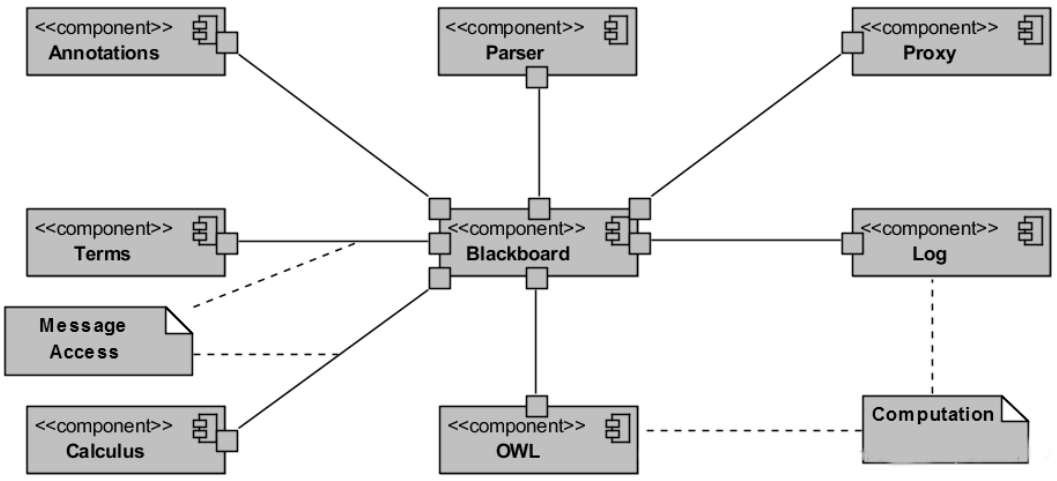


Figura 3-6 - Vista da arquitetura do motor de análise e avaliação.

Os requisitos não-funcionais RNF01 e RNF03, descritos na Tabela 3-3, foram o principal motivo da escolha deste padrão. Através dele é possível repartir o processo de análise e avaliação do ficheiro de metadados por um conjunto de componentes, cuja interação se efetua sobre uma base comum, sendo que cada um constitui um espaço único onde cada uma das tarefas é executada assincronamente sobre um único *Java Thread* do processo principal do motor. Isto garante que cada componente tem assim (*i*) um conjunto

de funcionalidade coesas (*high coheson*), i.e., uma lista de funcionalidades com uma forte relação entre si, (*ii*) uma fraca interdependência entre eles (*low-coupling*), i.e., para a execução das suas funcionalidades não necessita da invocação de nenhuma outra funcionalidade de nenhum outro componente.

Para a construção do motor foi identificado um conjunto de componentes, onde cada um tem assim uma função particular e claramente identificada no processo de análise e avaliação de um dado ficheiro de metadados. O seu modo de interação é por natureza sequencial na medida em que o processo de análise se desenvolve através de um conjunto de passos que exigem uma ordem temporal entre si. Do ponto de vista desta natureza um outro paradigma poderia ter sido utilizado, o padrão *PipeLine*, no entanto, tendo em conta o tipo de utilização deste motor, que no pior cenário poderá ser de acesso concorrencial por múltiplos utilizadores, foi considerado que o padrão *Blackboard* apresenta um melhor desempenho, pois permite que múltiplos pedidos concorrenciais possam ser respondidos sem a hipotética formação de uma fila de espera.

Assim foram definidos os seguintes componentes (descritos no diagrama de componentes na Figura 3-6), de modo a responder a um pedido de análise e avaliação de um ficheiro de metadados:

- **Componente *Blackboard***: Este componente tem a responsabilidade de manter os dados partilhados entre os restantes componentes. Através dele são trocadas múltiplas mensagens entre os componentes internos da arquitetura.

- **Componente *Proxy***: Este componente tem a responsabilidade da intermediação da comunicação entre o cliente que envia o pedido, e motor que faz a sua análise e avaliação. Estabelece-se como ponto de contato com todo o sistema através um protocolo, ou linguagem, pré-definida. Por ele passam assim as mensagens de entrada e saída do motor de análise e avaliação.

- **Componente *Parser***: Este componente tem como responsabilidade o controlo de todo o processo de análise de um dado ficheiro de metadados. Através dele é estabelecida uma ordem de trabalho, onde tanto o componente *Annotations*, como o componente *Terms*, têm de colaborar.

- **Componente *Annotations***: Este componente tem a responsabilidade de analisar sequencialmente todo o ficheiro de metadados e extrair todas as anotações com referência a conceitos ontológicos utilizados, que estiverem de acordo com o extrator a ser utilizado. Este extrator é explicitamente

estabelecido na chamada do cliente, indicando assim que estrutura lógica do ficheiro deverá ser considerada.

- **Componente *Terms***: Este componente tem como responsabilidade a análise sequencial do ficheiro de metadados e extração da descrição de todas as anotações de modo geral, i.e., anotações com, e sem, referência a conceitos ontológicos, que possam estar de acordo com o extrator a ser utilizado, à semelhança do componente anterior.

- **Componente *Calculus***: Uma vez encontradas todas as anotações, com ou sem referência a conceitos ontológicos, este componente tem a responsabilidade de controlar o processo de avaliação global dos metadados. Terminado este processo é calculada a média de especificidade e cobertura de cada uma das classes de anotação existentes e do estudo em causa.

- **Componente OWL**: Cabe a este componente a responsabilidade de interação com o repositório de dados, de modo a determinar o valor individual de especificidade de um determinado termo ontológico.

- **Componente Log**: Este componente tem como responsabilidade o registo de todas as ações de registo, evocados pelos componentes do motor.

3.2.3.1 Componente *Blackboard*

Este é o componente chave de toda a arquitetura. Através da implementação do conceito de espaço de Tuplos (uma coleção de Tuplos onde cada um representa uma lista finita de elementos, com diferentes tipos de dados), por sua vez uma implementação do paradigma de memória associativa para computação distribuída que proporciona um repositório de Tuplos [75], implementa o padrão de desenho *Blackboard* (explicado anteriormente e ilustrado na Figura 3-7), que permite a cada componente do motor enviar e receber mensagens assincronamente, de e para outros componentes, com recurso a um conjunto de operações (escrita, leitura com e sem bloqueio de Tuplo, leitura e eliminação do repositório com e sem bloqueio de Tuplo), ilustradas na Figura 3-7, de modo a resolver o "problema" de análise e avaliação de um ficheiro de metadados.

No arranque do motor de análise e avaliação é estabelecida uma instância em memória deste componente e partilhada através do paradigma de *Dependency Injection*, um padrão de desenho de *software* que utiliza a inversão de controlo de modo a resolver as dependências, com os restantes componentes. Nesta implementação do espaço de Tuplos, no componente *Blackboard*, cada Tuplo representa uma lista ordenada em memória de dois elementos (*2-tuple*), representado por um par {Chave-Valor}, onde o

elemento Chave identifica o componente ao qual se destina a mensagem e o elemento Valor salvaguarda a mensagem propriamente dita. Cabe a cada componente observar o estado da memória partilhada e averiguar em tempo-real se existe algum pedido sobre o qual ele pode desempenhar algum processo.

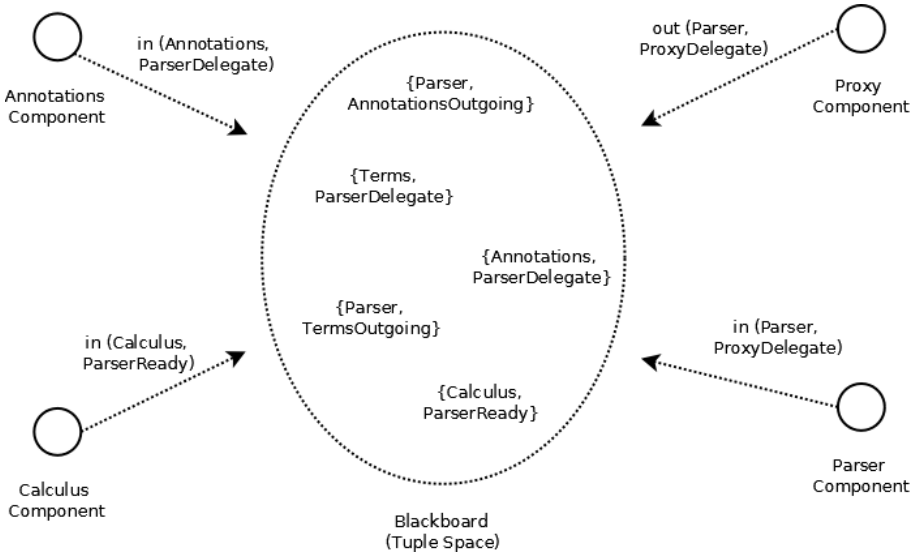


Figura 3-7 - Espaço de Tuplos, utilizado pelo componente Blackboard

Em caso afirmativo, o Tuplo é lido pelo componente e imediatamente eliminado do espaço de Tuplos. Cada componente coloca ou retira Tuplos da memória partilhada, de acordo com aquelas que são as suas funções e de acordo com aquele que é o resultado do seu processamento sobre o procedimento em curso, de análise e avaliação de metadados.

Tabela 3-4 - Protocolo padrão de uma mensagem interna ao MAA.

Campo	**Descrição**
Timestamp	Grupo data-hora de envio da mensagem
UniqueID	Identificador único da mensagem
Target	Componente destinatário da mensagem

A nomenclatura utilizada no elemento Chave é simples. O valor a ser colocado na chave é uma representação do nome do componente a quem é destinada a mensagem, juntamente com o sufixo de direccionalidade da mesma, i.e., de entrada (*In*) ou saída (*Out*). Deste modo existem: (*i*) *Proxy(In|Out)*, (*ii*) *Parse(In|Out)*, (*iii*) *Annotations(In|Out)*, (*iv*) *Terms(In|Out)*, (*v*) *Log(In)*, (*vi*) *Calculus(In|Out)*, (*vii*) *OWL(In|Out)*. Existe ainda um outro valor, o *Digest*, que difere dos restantes na medida em que é apenas utilizado para propagar uma mensagem de progresso do processo geral em desenvolvimento. Pode ser enviada por qualquer componente, mas o seu único

destinatário é o componente Proxy, que a utiliza para informar o cliente que inicialmente fez o pedido.

Tabela 3-5 - Lista de mensagens trocadas entre componentes do MAA.

Componente	**Execução**	**Descrição**
Proxy	*Delegate*	Enviada após a chegada de um novo pedido
Parser	*Ready*	Enviada após a conclusão do processo de análise
	Delegate	Enviada para sub-rotina dos componentes *Annotations* e *Terms*
Annotations	*Outgoing*	Enviada após a conclusão da sub-rotina de análise de anotações
Terms	*Outgoing*	Enviada após a conclusão da sub-rotina de análise de termos
Calculus	*Ready*	Enviada após a conclusão do processo de avaliação do ficheiro de metadados
	Delegate	Enviada para avaliação de uma anotação em particular
OWL	*Outgoing*	Enviada após a avaliação de uma anotação

A nomenclatura utilizada para o elemento Valor segue a implementação base de um protocolo pré-estabelecido de mensagens internas, descrito na Tabela 3-4. Dada a complexidade das mensagens trocadas entre componentes do motor, foi necessário definir uma estrutura particular a cada um. Estas mensagens englobam naturalmente elementos comuns, mas diferem sobretudo no corpo. Deste modo foram definidos quatro momentos de execução de um componente, de modo a englobar os vários tipos de mensagens: (*i*) *Ready*, (*ii*) *Delegate*, (*iii*) *Outgoing* e (*iv*) *Ingoing*:

- ***Ready***: Indica mensagens que foram enviadas após a conclusão processamento de uma tarefa principal, pelo componente.

- ***Delegate***: Indica mensagens que delegam sub-rotinas de processamento de tarefas principais, a um componente.

- ***Outgoing***: Representa mensagens que são enviadas após a finalização de processamento de sub-rotinas, em resposta a mensagens do tipo *Delegate*.

- **Ingoing**: Representa mensagens de chegada, apenas para o componente Log.

Como descrito na Tabela 3-5 e ilustrado na Figura 3-8, o componente Proxy envia uma mensagem do tipo *Delegate* ao componente Parser, a delegar a execução de um processo de análise e avaliação para os metadados recebidos. O componente *Parser*

delega, por sua vez, a avaliação dos metadados aos componentes *Annotations* e *Terms*, através do envio de uma mensagem do tipo *Delegate*. Cada um destes, após a conclusão da sua tarefa, envia os resultados obtidos ao componente *Parser*, com recurso a uma mensagem do tipo *Outgoing*. Este por sua vez dá por completa a tarefa de análise dos metadados e envia os resultados ao componente *Calculus*, com auxílio de uma mensagem do tipo *Ready*. O componente *Calculus* delega no componente OWL, através do envio de uma mensagem do tipo *Delegate*, a determinação dos valores de especificidade de cada um dos termos semânticos encontrados pelo processamento feito pelo componente *Parser*. No final do processo de avaliação, o componente *Calculus* envia os resultados ao componente *Proxy*, através de uma mensagem do tipo *Ready*, para que este possa informar o cliente dos resultados obtidos pelo motor.

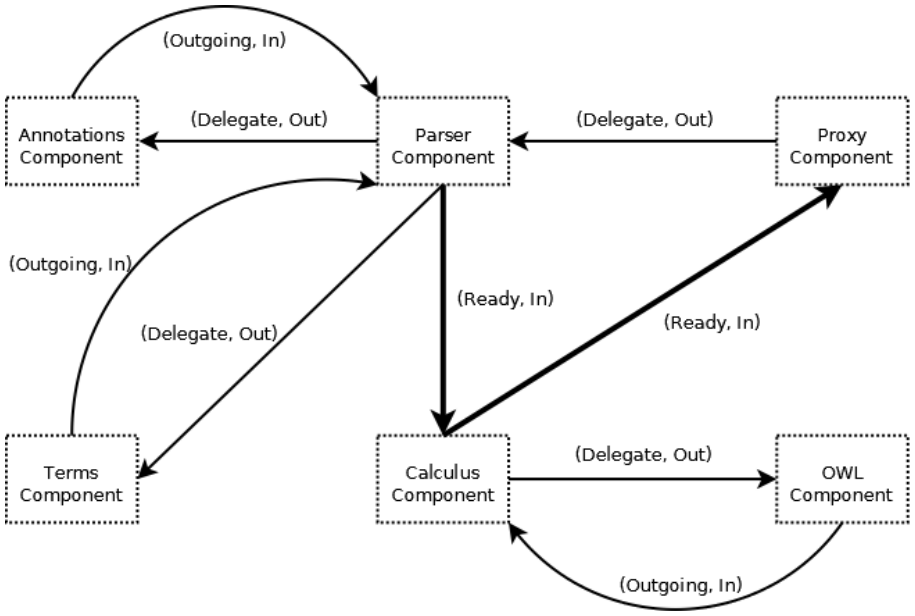


Figura 3-8 - Fluxo, tipo e direcionalidade de mensagens do MAA

3.2.3.2 Componente *Proxy*

Este componente apresenta-se como a porta de entrada para todo o sistema de análise e avaliação. É o comutador de mensagens entre os clientes externos e os componentes internos da arquitetura, que o utilizam quer para (*i*) solicitar uma prestação de serviços, quer para (*ii*) receber informação sobre a progressão do trabalho a ser efetuado, ou de algum problema que tenha sido encontrado durante esta. Permite que todo o sistema possa ser considerado como uma caixa preta, com um único ponto de entrada de parâmetros e de saída de resultados. Existem assim dois grandes fluxos de informação, um de entrada de informação na plataforma, outro de saída, todos eles processados através deste componente.

No arranque do sistema este componente, para além de receber uma referência da instância do componente de *Blackboard* que lhe permite manter comunicação com os restantes componentes do motor, estabelece um porto TCP sobre o endereço IP, indicado no arranque do motor, de escuta a novos pedidos por parte dos clientes externos através de um *Java Socket*, detalhado na Secção 2.5.2. Estes podem estar localmente situados no mesmo anfitrião, ou em um qualquer sítio remoto com acesso à *Internet*. Sempre que chega uma nova mensagem externa é desencadeado um evento de análise, que tem por último fim dar início a mais um processo interno de análise e avaliação. Em sentido inverso, sempre que chega uma mensagem interna é encontrado o destinatário externo e enviada a mensagem respetiva.

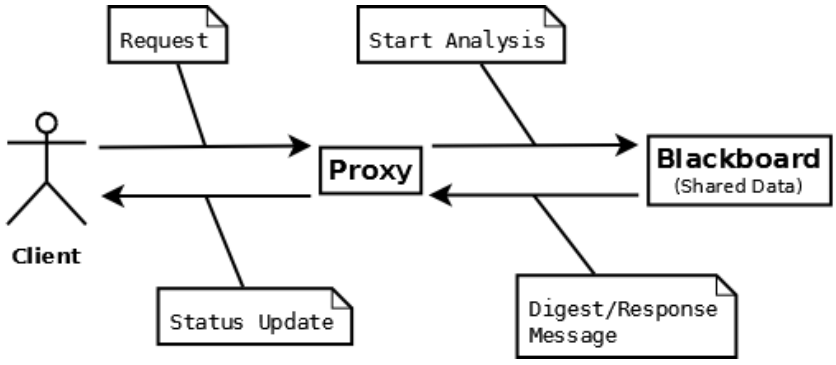


Figura 3-9 - Fluxos de informação existente no componente Proxy do MAA.

Os fluxos de informação, de entrada e saída da plataforma, representados na Figura 3-9, são geridos pelo componente Proxy de modo assíncrono, i.e., a estrutura interna deste objeto foi desenvolvida de modo a que várias linhas de processamento consigam ser executadas em paralelo, sem necessidade de momentos de espera em qualquer um dos sentidos. No entanto, e por necessidade de acesso a zonas partilhadas de memória, foram escolhidos objetos cuja natureza assenta na existência de zonas de mútua exclusão, i.e., zonas de memória onde dois ou mais objetos necessitem acesso concorrencial. Por ter sido implementado em *Java*, dois desses objetos são o *ConcurrentHashMap* e a *LinkedBlockingQueue*.

Tabela 3-6 - Protocolo de mensagem TCP do componente Proxy do MAA.

Campo	**Descrição**
UUID	Identificador único da mensagem
SenderTCPIP	Endereço IP do emissor da mensagem
SenderTCPPort	Porta TCP do emissor da mensagem
ReceiverTCPIP	Endereço IP do recetor da mensagem
ReceiverTCPPort	Porta TCP do recetor da mensagem

Timestamp	Grupo data-hora de envio da mensagem
Type	Tipo de mensagem (Enumeração de tipos de mensagem)
Body	Corpo da mensagem em formato binário
TargetRepository	Indentificador de repositório sobre o qual a análise será feita

Após a chegada de uma mensagem TCP externa esta é descodificada e convertida para uma instância interna do protocolo utilizado na sua transmissão (ilustrado na Figura 3-10 a), etapa 1). Este protocolo estabelece a estrutura de uma mensagem, detalhada na Tabela 3-6, a partir da qual é extraído um conjunto de informações de modo a dar início ao processo geral. Por cada mensagem externa recebida, este componente salvaguarda na memória partilhada por todos os componentes do motor de análise e avaliação uma série de elementos que identificam de modo único o cliente e o respetivo pedido.

Tabela 3-7 - Estrutura em memória de um pedido no componente Proxy do MAA.

Campo	**Descrição**
RequestUUID	Identificador único da mensagem de origem
SenderTCPIP	Endereço IP do emissor da mensagem
SenderTCPPort	Porta TCP do emissor da mensagem
SentTimestamp	Grupo data-hora (*Epoch*) de envio da mensagem
ReceivedTimestamp	Grupo data-hora (*Epoch*) de receção da mensagem
RequestType	Identificador do tipo de pedido feito

Para tal, é gerado um identificador único do pedido (*UUID*, um valor alfanumérico de 128-bits), ilustrado na Figura 3-10 a), etapa 2, utilizado como chave dessa estrutura em memória (descrita na Tabela 3-7) onde são colocados os elementos principais da mensagem TCP recebida (ilustrado na Figura 3-10 a), etapa 3): (*i*) identificador único (*RequestUUID*) e (*ii*) grupo data-hora de envio (*SentTimestamp*); assim como elementos de identificação do nó de rede: o endereço IP (*SenderTCPIP*) e a porta TCP (*SenderTCPPort*). Este identificador reveste-se de vital importância para todo o processo, pois estabelece a fronteira lógica de processamento paralelo a ser executado por todo o sistema.

Depois de criada esta estrutura e salvaguardada em memória (ilustrado na Figura 3-10 a), etapa 4), é lido o repositório indicado pelo cliente na mensagem TCP/IP (*TargetRepository*, na Tabela 3-6), sobre o qual todo o processo se desenrolará. Após a recolha do corpo da mensagem (*Body*, na Tabela 3-6), dá-se início ao processo interno geral de análise e avaliação através de uma mensagem do tipo *ProxyDelegate* (Tabela 3-5) dirigida ao componente *Parser* (ilustrado na Figura 3-6), enviada através do

componente *Blackboard*, onde consta (*i*) o identificador único do pedido, (*ii*) o identificador do repositório de metadados e (*iii*) o ficheiro a ser analisado em formato binário.

Em sentido inverso, após a chegada de uma mensagem interna ao componente é recuperado o identificar único do pedido (*UUID*) e encontrada em memória a sua estrutura individual, indicada na Tabela 3-7 (ilustrado na Figura 3-10 b), etapas 1, 2 e 3, respetivamente). Com base no endereço IP (*SenderTCPIP*) e porta TCP (*SenderTCPPort*) é estabelecida uma nova comunicação TCP/IP e enviada uma mensagem ao cliente com base no protocolo pré-estabelecido, detalhado na Tabela 3-6 (ilustrado na Figura 3-10 b), etapa 4). Apenas dois tipos de mensagens internas são recebidas pelo componente *Proxy*: (*i*) *Digest* e (*ii*) *CalculusReadyOutgoing*. A primeira identifica uma mensagem de evolução do processo de avaliação, a segunda uma mensagem com o resultado final do mesmo.

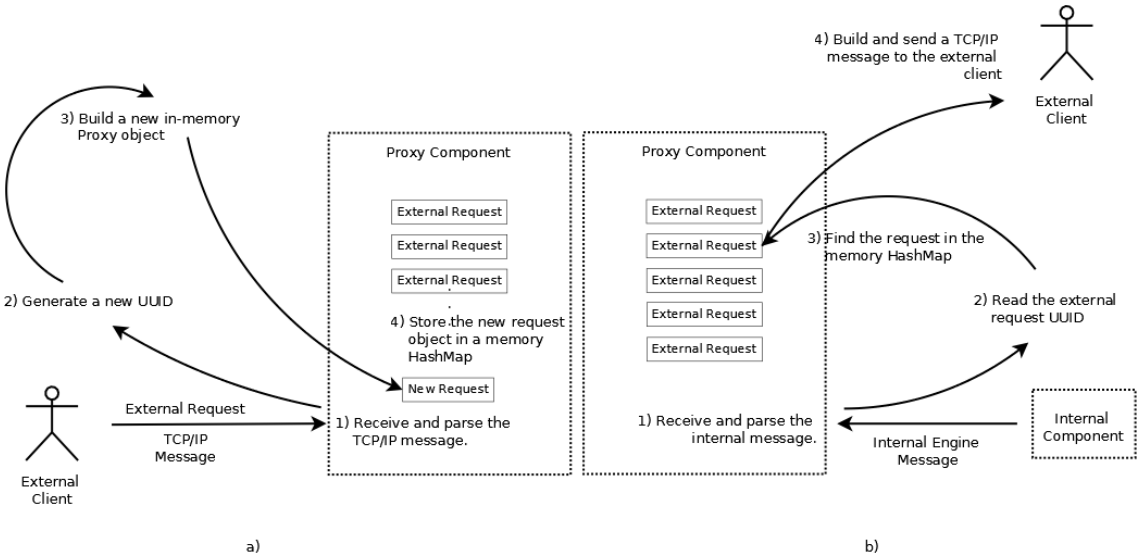


Figura 3-10 - Fluxo de operação interna do compoennte Proxy do MAA.

As mensagens *Digest* são enviadas por todos os componentes da arquitetura, à exceção do componente *Log*, que têm intervenção direta no processo de análise e avaliação. As mensagens *CalculusReadyOutgoing* são apenas enviadas pelo componente *Calculus*, como o prefixo indica, após a conclusão do processo de avaliação do ficheiro de metadados.

3.2.3.3 Componente *Parser*

O componente *Parser* tem como principal objetivo o controlo do procedimento de análise (PAN) de um ficheiro de metadados, enviado pelo cliente. Para levar a cabo este processo, o componente recorre aos serviços de outros dois componentes do motor, (*i*) o componente *Annotations* e (*ii*) o componente *Terms*. Para tal, define em memória uma estrutura que lhe permite manter um registo de processos ativos e respetivo progresso.

Cada um destes processos é iniciado com a receção de uma mensagem *ProxyDelegate*, enviada por parte do componente Proxy (Figura 3-11, etapa 1.1.1). Baseada no protocolo base da Tabela 3-4,acresce a esta estrutura o corpo da mensagem, o ficheiro de metadados, que nesta altura ainda se encontra em formato binário, representado por um vetor de *bytes*.

Tabela 3-8 - Estrutura de controlo do PAN do componente Parser do MAA.

Campo	**Descrição**
JobID	Identificador único do processo geral
MetaData	Representação do resultado final da solução
ParseStatus	Enumeração do estado de processamento
RequestType	Identificação do tipo de requisição

A cada pedido, recebido através do componente *Blackboard*, corresponde uma entrada na estrutura de controlo em memória. A cada entrada corresponde, por sua vez, um par {Chave-Valor}, onde a Chave é o identificador único do pedido (*UUID*) e o Valor uma subestrutura que regista a identificação do processo, do resultado do mesmo e respetivo estado de progresso, descrita na Tabela 3-8. O campo *JobID* identifica unicamente o processo de avaliação interno ao componente, o campo MetaData representa uma estrutura inicial de resposta do motor de análise e avaliação ao pedido do cliente e o campo *ParseStatus* regista o último estado do procedimento, tendo em conta o trabalho desempenhado pelos componentes *Annotations* e *Terms*.

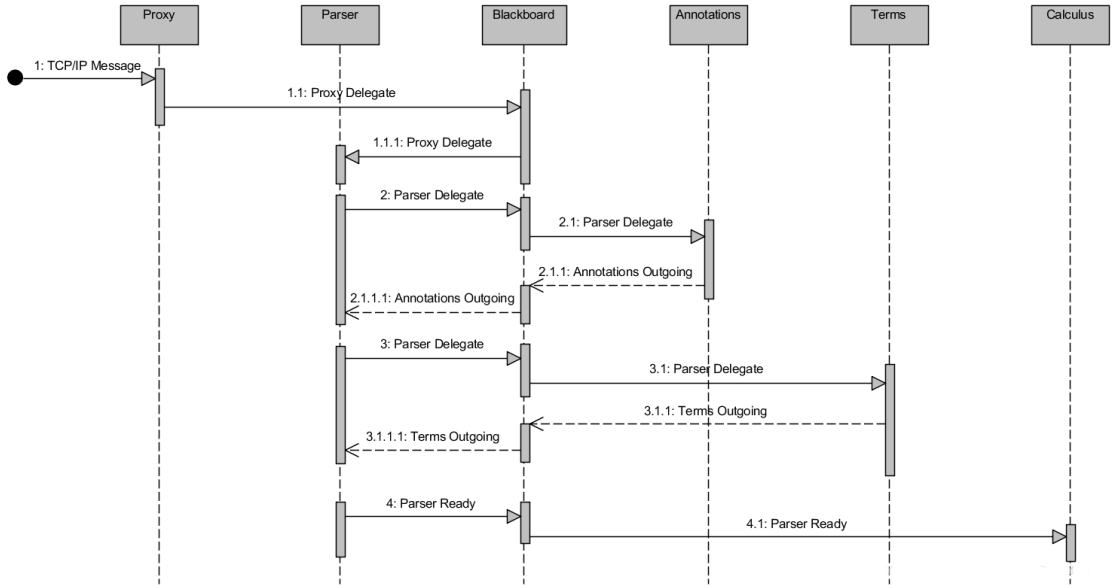


Figura 3-11 - Etapas do PAN controlado pelo componente Parser do MAA.

O procedimento de análise compreende, de modo geral, três etapas:

1. **Construção do cabeçalho**: Extração de informação geral do ficheiro de metadados, assim como da lista de ontologias referidas. Primeira instanciação da estrutura lógica de resposta dada pelo motor de análise e avaliação ao cliente.

2. **Construção das listas de anotações**: Extração da lista de anotações existentes, i.e., com e sem referência a conceitos ontológicos.

3. **Conclusão do procedimento**: Conclusão do procedimento e posterior envio do processo ao próximo componente.

Construção do cabeçalho

O objetivo do primeiro passo do procedimento de análise é obter um conjunto de informações de caráter generalista. Torna-se necessário obter dados tais como o identificador de estudo e a lista de ontologias utilizadas na referenciação de anotações. No entanto, e de modo a atingir este objetivo, é mandatório conhecer de antemão qual a estrutura de um ficheiro de metadados. Cada repositório público segue a sua nomenclatura de definição de ficheiros de metadados, para a informação que contêm, no entanto, no âmbito desta contribuição foi considerado que os metadados são por norma definidos por múltiplas secções de anotação, aqui denominadas de classes, onde em cada uma se encontra definido um conjunto de anotações, das quais algumas se encontram associadas a conceitos ontológicos e outras não.

Para resolver este desafio foi implementado na estrutura OOP o padrão de desenho *Adapter*, descrito em [76], que permite ao seu utilizador a abstração do detalhe de implementação de um conjunto de funcionalidades, i.e., aquilo que é conhecido é apenas um conjunto de ações que são implementadas sobre um número de circunstâncias completamente ortogonais. Para esse efeito foram definidos quatro interfaces, enumerados na Tabela 3-9: (*i*) *MetaHeader*, (*ii*) *MetaOntologies*, (*iii*) *MetaAnnotations* e (*iv*) *MetaTerms*.

Tabela 3-9 - Lista de ações dos interfaces para extração de informação.

Interface	Ação	Descrição
MetaHeader	*GetStudyID*	Obter o identificar geral do estudo sobre o qual reportam os metadados
MetaOntologies	*GetMetaOntologies*	Obter lista das ontologias identificadas nos metadados

MetaAnnotations	*GetMetaAnnotations*	Obter lista das anotações presentes numa determinada classe
MetaTerms	*GetMetaTerms*	Obter lista de termos presentes numa determinada classe

A interface *MetaHeader* tem como principal função a captura do identificador do estudo. A interface *MetaOntologies* tem como objetivo encontrar todas as ontologias a que o ficheiro de metadados faz referência. A interface *MetaAnnotations* tenta recuperar todas as anotações existentes, i.e., tenta encontrar descrições como: *http://host-name/ontology/concept*. A interface *MetaTerms* obtêm uma lista de termos presentes no ficheiro.

Com primeira ação, denominada *MetaData*, inicia-se a tomada de forma da resposta a enviar ao cliente do motor. Implementada através do paradigma *Object-Oriented*, detalhado na 2.5.2, representa aquilo que é considerado essencial para o objetivo final da solução. Esta ação (ilustrada no diagrama de classes da Figura 3-12) compreende um conjunto de propriedades que tentam refletir, de modo geral, o que se pondera ser um ficheiro de metadados genérico. O seu desenho baseou-se na especificação ISA-TAB, descrita na Secção 2.4.1, de modo a capturar a complexidade utilizada na interpretação de resultados de experiências, combinando várias tecnologias.

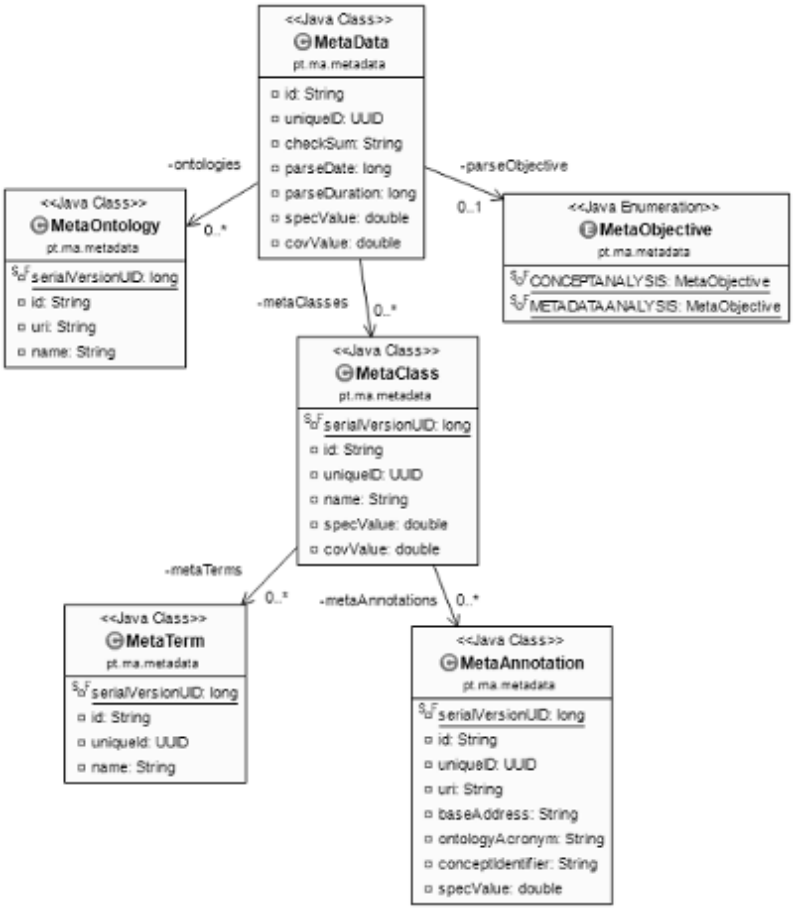


Figura 3-12 - Diagrama de classes da resposta base dada pelo MAA.

A estrutura é composta por um identificador único que geralmente é indicado em cada estudo ou ação de investigação, um conjunto de ontologias de referência, um conjunto de classes de estudo e as médias de especificidade e cobertura que se pretendem calcular. Quer as ontologias, quer as classes, são elas próprias subestruturas (as classes *MetaClass* e *MetaOntology*, na Figura 3-12) de *MetaData*, com um conjunto distinto de propriedades. Cada classe, para além das propriedades que a identificam, agrega as anotações, com e sem referências a conceitos ontológicos encontrados em cada uma (classes *MetaAnnotations e MetaTerms* respetivamente, na Figura 3-12), para além da média de especificidade e cobertura a calcular (campos *SpecValue* e *CovValue, na* Figura 3-12). Por sua vez, cada referência a um termo ontológico compreende o valor de endereçamento único, o URI, e o valor de especificidade que se pretende determinar, com base na sua ontologia de referência. Cada anotação salvaguarda a descrição encontrada.

Construção das listas de anotações e termos

No decorrer do procedimento de análise, depois de recebida a mensagem e feito o registo de um novo processo (Figura 3-11, etapa 1.1.1), este componente transmite uma mensagem ao componente *Annotations* e outra ao componente *Terms* (Figura 3-11, etapas 2 e 3, respetivamente). Contida em ambas as mensagens encontra-se uma delegação de tarefas, de modo a que sobre o ficheiro de metadados enviado seja feita uma análise com o sentido de extrair, quer as anotações, quer os termos nele contido. A mensagem é do tipo *ParseDelegate*, descrito na Tabela 3-5, e nela constam o repositório base de análise e o ficheiro, ainda em modo binário.

À semelhança de outros componentes, o modo de execução deste componente é também ele assíncrono. Enquanto aguarda pela resposta dos componentes a quem requisitou uma ação, fica disponível para receber novos pedidos. Porque o tempo de execução dos componentes é naturalmente diferente, o procedimento de espera decorre em uma linha de execução paralela. Durante o decorrer de cada procedimento o valor da propriedade *ParseStatus*, descrita na Tabela 3-8, será atualizado de modo a refletir o progresso do mesmo.

Conclusão do procedimento

Assim que ambos os componentes terminam a sub-rotina delegada, e enviam de volta o resultado da sua execução (Figura 3-11, etapas 2.1.1.1 e 3.1.1.1), o componente *Parser* dá o procedimento de análise por completo e envia, por sua vez, o resultado ao componente *Calculus* (etapa 4 do procedimento de análise, na Figura 3-11). Caberá ao último a execução do procedimento de avaliação daquilo que foi encontrado. A mensagem enviada é do tipo *ParseReady*, descrita na Tabela 3-5, e nela consta já uma

instância do resultado final (*MetaData*, ilustrado no diagrama da Figura 3-12). No final da execução do procedimento, este é retirado da lista em memória.

3.2.3.4 Componente *Annotations*

O componente *Annotations* é um dos componentes sobre os quais o *Parser* delega sub-rotinas de processamento, no procedimento de análise do motor. Este componente tem a responsabilidade de encontrar e extrair anotações com referência a conceitos ontológicos existentes no ficheiro de metadados de modo a completar a etapa 2 do procedimento de análise, na Figura 3-11. Recebe uma mensagem da qual faz parte o identificador do repositório de metadados, uma coleção de classes do tipo *MetaClass*, ilustrada na Figura 3-12, na qual serão salvaguardadas as anotações encontradas e o ficheiro original de metadados, em formato binário, submetido ao motor.

Com base no repositório indicado será feita a escolha do adaptador para análise do ficheiro. As funcionalidades a executar serão aquelas oferecidas pela interface *MetaAnnotations*, descrito na Tabela 3-9. Por cada uma das classes de anotação dos metadados, são recolhidas todas as anotações com referência a conceitos ontológicos encontradas e enquadradas posteriormente na estrutura *MetaData*, ilustrada no diagrama de classes da Figura 3-12. A cada uma destas anotações corresponde um URI, conceito descrito na Secção 2.1.1, à qual corresponderá um conceito, ou termo, de uma ontologia, ou vocabulário, externo, público e aceite por toda a comunidade, p. ex., a anotação *http://www.ebi.ac.uk/efo/EFO_0000400*, que corresponde ao termo *"diabetes mellitus"*, da ontologia *Experimental Factor Ontology*[7] (EFO).

Uma vez completada a tarefa, este componente devolve o resultado ao componente *Parser* (como demostrado no diagrama de iteração da Figura 3-11, etapa 2.1.1), através de uma mensagem do tipo *AnnotationsOutgoing*, da qual constam o identificador único do pedido e um conjunto de classes de metadados preenchidas com as anotações encontradas.

3.2.3.5 Componente *Terms*

O componente *Terms* é um dos componentes sobre os quais o *Parser* delega sub-rotinas de processamento. Este componente tem a responsabilidade de encontrar e extrair a descrição das anotações utilizadas no ficheiro de metadados, na descrição do conjunto de dados, de modo a completar a etapa 3 do procedimento de análise (Figura 3-11). Recebe uma mensagem da qual faz parte o identificador do repositório de metadados, uma coleção de classes do tipo *MetaClass*, ilustrado na Figura 3-12, nas quais serão

[7] http://www.ebi.ac.uk/efo

salvaguardas as anotações e o ficheiro original de metadados, em formato binário, submetido ao motor.

À semelhança do componente *Annotations*, com base no repositório indicado será feita a escolha do adaptador para análise do ficheiro. O conjunto de funcionalidades a executar serão aquelas oferecidas pela interface *MetaTerms*, enumerado na Tabela 3-9. Por cada uma das classes são recolhidas todos os termos encontrados e enquadrados na estrutura ilustrada no diagrama de classes da Figura 3-12. Uma vez completada a tarefa devolve o resultado ao componente *Parser* (como demostrado no diagrama de iteração da Figura 3-11, etapa 3.1.1), através de uma mensagem do tipo *TermsOutgoing*, da qual constam o identificador único do pedido e um conjunto de classes de metadados preenchidas com os termos encontrados.

3.2.3.6 Componente *Calculus*

O componente *Calculus* tem como objetivo o controlo do segundo procedimento do motor de análise e avaliação, o procedimento de avaliação (PAV), i.e., a avaliação do valor de especificidade de cada uma das anotações encontradas, com referência a conceitos ontológicos, a razão de cobertura entre anotações com e sem referência a conceitos ontológicos e cálculo de médias globais para cada uma das classes de anotação e estudo final, no qual estas se inserem. Por intermédio do componente *Blackboard*, o componente *Calculus* recebe uma mensagem do tipo *ParserReady*, detalhada na Tabela 3-5, enviada pelo componente *Parser* como o prefixo indica, após o fim do procedimento de análise controlado por este. Esta mensagem acresce à mensagem tipo do sistema, descrita na Tabela 3-4, o corpo que nesta altura do processamento apresenta já a estrutura do resultado final a entregar ao cliente (*MetaData*), ilustrada no diagrama de classes da Figura 3-12.

À semelhança do componente *Parser*, estabelece uma estrutura em memória para controlo e registo dos pedidos que lhe são endereçados. Esta apresenta uma disposição de {Chave-Valor}, em que a Chave identifica de modo único uma subestrutura (detalhada na Tabela 3-10) cujas propriedades permitem identificar unicamente o pedido feito ao componente (*JobID*), em que ponto se encontra o seu progresso (*TaskSet*) e qual o seu resultado (*MetaData*). Sempre que uma mensagem é recebida, com origem no componente *Parser*, é criada uma nova entrada nesta estrutura.

Tabela 3-10 - Estrutura de controlo do PAV do componente Calculus.

Campo	Descrição
JobID	Identificador único do processo de avaliação
MetaData	Representação do resultado final da solução

JobTaskList	Lista de sub-rotinas da própria tarefa, cada uma corresponde a uma chamada ao componente OWL
TaskSet	Indicador de conclusão da tarefa, valor booleano
RequestType	Enumeração do tipo de requisição

Acresce a esta estrutura uma lista de sub-rotinas de cada uma das tarefas do componente. Por cada uma, e por cada classe de metadados a avaliar, uma nova entrada será colocada na propriedade *JobTaskList* (). Esta segue também uma estrutura de {Chave-Valor}, onde a chave será o identificador único da classe (*UUID*) e o valor uma instância da representação de classe, da estrutura de resultados do motor de análise e avaliação (ilustrada no diagrama de classes da Figura 3-12).

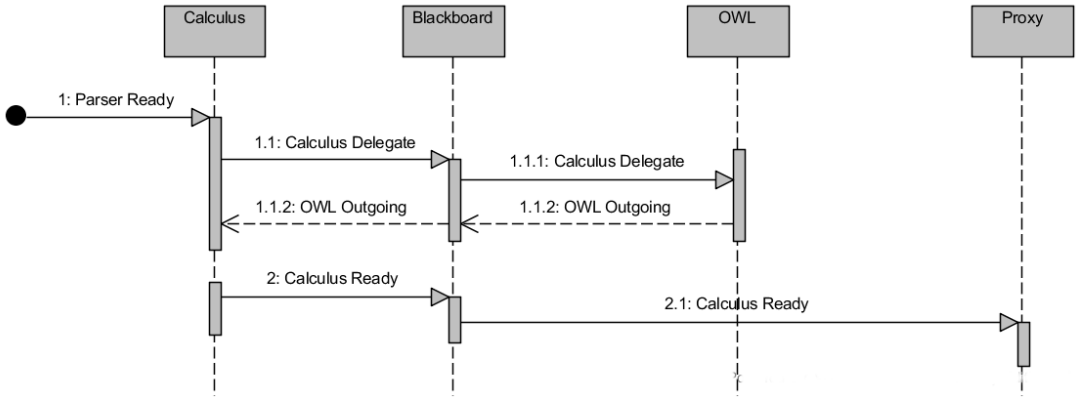


Figura 3-13 - Etapas do PAV controlado pelo componente Calculus.

O procedimento de avaliação é composto, de modo geral, por quatro etapas:

1. **Avaliação de anotações**: Avaliação individual do valor de especificidade de cada uma das anotações com referência a conceitos ontológicos, presentes nas classes contidas na mensagem.

2. **Cálculo de médias das classes**: Cálculo da média aritmética do valor de especificidade e de taxa de cobertura de cada uma das classes.

3. **Cálculo de médias globais**: Cálculo da média aritmética do valor de especificidade e cobertura do ficheiro inteiro.

4. **Conclusão do procedimento**: Envio do resultado final ao componente Proxy e execução de rotinas de limpeza de memória.

Avaliação das anotações

O objetivo da primeira etapa do procedimento é obter um valor quantificável cujo domínio se encontra entre [0, 1], de especificidade de cada uma das anotações com referência a conceitos ontológicos no ficheiro de metadados. Um pormenor importante, para as anotações que não seja possível calcular um valor de especificidade, este toma o valor de -1. Para cumprir com este objetivo necessita de delegar a tarefa de quantificação sobre um outro componente, o componente *OWL*, ilustrado no diagrama de componentes da Figura 3-6.

```
{
    "ontologies": [
        "bao",
        "bto",
        "chmo",
        "cmo",
        "efo",
        "envo",
        "go",
        "hl7",
        "mp",
        "ms",
        "nemo",
        "obi",
        "pato",
        "po",
        "sdon",
        "sbo",
        "uo",
        "xco"
    ]
}
```

Figura 3-14 - Representação do ficheiro de configuração do componente OWL.

Depois de estabelecer uma nova entrada na estrutura de controlo de tarefas, e para cada uma das classes existentes na estrutura recebida pelo componente *Parser*, uma mensagem do tipo *CalculusDelegate* é enviada ao componente *OWL*, ilustrada no diagrama de iteração da Figura 3-13 (etapa 1.1 do procedimento de avaliação). Até ao momento de conclusão de todas as sub-rotinas presentes na propriedade *JobTaskList*, i.e., até que todas as classes tenham as suas anotações avaliadas, esta etapa continuará em execução. Depois de todos os resultados terem sido obtidos do componente *OWL* dá-se início à segunda etapa do procedimento.

Cálculo de médias das classes

Na segunda etapa é calculada a média de especificidade e a razão de cobertura de cada classe, tendo em conta as medidas de avaliação de qualidade de integração semântica descritas na Secção 3.1. Para tal, consideram-se todas as anotações que tenham sido avaliadas com sucesso por classe, i.e., cujo valor de especificidade se encontre entre [0, 1]. Todas as outras obtêm o valor de -1. Os valores calculados são salvaguardados nos campos *SpecValue* e *CovValue* do objeto *MetaClass* da classe de anotação correspondente, da estrutura de resposta do motor de análise e avaliação, representada no

diagrama ilustrado na Figura 3-12, armazenada na estrutura interna do componente, descrita na Tabela 3-10.

Cálculo das médias globais

A terceira etapa tem objetivos semelhantes aos da etapa anterior, i.e., cálculo da média de especificidade das anotações com referência a conceitos ontológicos e taxa de cobertura, desta feita tendo em conta todo o ficheiro de metadados, i.e., é efetuada a avaliação de qualidade de integração semântica dos metadados utilizados, referida na Secção 3.1. São assim consideradas todas as anotações para as quais foi possível calcular um valor de especificidade, para o cálculo da média do ficheiro de metadados. Este valor é salvaguardado no campo *SpecValue* do objeto *MetaData*, representado no diagrama de classes da Figura 3-12. Para cálculo da média de cobertura do ficheiro de metadados são consideradas as anotações com referência a conceitos ontológicos e o número total de anotações utilizados na descrição dos metadados. O valor encontrado é salvaguardado no campo *CovValue* do objeto referido anteriormente. Este por sua vez é armazenado na estrutura interna do componente, descrita na Tabela 3-10.

Conclusão do procedimento

Na quarta e última etapa do procedimento de avaliação, após a salvaguarda dos valores de média de especificidade e de taxa de cobertura de cada uma das classes existentes, assim como a média de especificidade e taxa de cobertura do todo o ficheiro de metadados na estrutura de resposta que tem vindo a ser construída, ilustrada no diagrama de classes da Figura 3-12, é construída uma mensagem do tipo *CalculusReady*, descrita na Tabela 3-5, da qual consta o resultado dos procedimentos de análise e de avaliação colocado na estrutura de resposta geral do motor, e enviada ao componente *Proxy* (etapas 2 do procedimento de avaliação, na Figura 3-13).

Depois de enviada, é eliminada da estrutura de controlo de progresso do componente a instância relativa a esta tarefa, descrita na Tabela 3-10.

3.2.3.7 Componente *OWL*

Cabe ao componente *OWL* obter o valor de especificidade das anotações com referência a conceitos ontológicos, de cada uma das classes que lhe são enviadas para avaliação através da estrutura representada no diagrama de classes da Figura 3-12. Recebe mensagens do componente *Calculus* e por cada uma deverá enquadrar as anotações existentes em uma das ontologias presentes no repositório. Assim por cada anotação o procedimento de avaliação segue agora as seguintes etapas:

1. **Enquadramento da anotação**: Análise do URI da anotação de modo a identificar qual a ontologia a que pertence.

2. **Verificação no sistema de cache**: Verificação no sistema de cache interno da existência de um valor de especificidade para a anotação.

3. **Cálculo de especificidade no repositório de ontologias**: Chamada ao repositório de ontologias para cálculo da especificidade da anotação.

4. **Conclusão do procedimento de avaliação**: Conclusão do procedimento de avaliação, através do envio de resultados.

Enquadramento da anotação

O enquadramento de anotações é feito através da comparação de um conjunto de acrónimos de ontologias com os endereços públicos, únicos a cada anotação. Estes acrónimos são lidos, no arranque do motor, a partir de um ficheiro (*ontology-list.json*) localizado na diretoria de configurações (*/configfiles*). O formato do ficheiro é JSON, descrito na Secção 2.5.5, e representa um vetor alfanumérico de acrónimos de ontologias, ilustrado na Figura 3-14.

Tabela 3-11 - Decomposição do endereçamento de um termo de Ontologia.

Campo	**Descrição**
URI	http://www.ebi.ac.uk/efo/EFO_0000400
Nome da Ontologia	*Experimental Factor Ontology*
Nome do conceito	*diabetes mellitus*
Protocolo	http
Anfitrião	www.ebi.ac.uk
Acrónimo	EFO
Identificação do termo	EFO_0000400

Esta comparação torna-se possível pela nomenclatura geralmente utilizada no endereçamento de um termo, de uma determinada ontologia. Esta segue a norma de endereçamento URI, onde é necessário indicar (*i*) o protoloco de comunicação (p. ex., HTTP ou HTTPS), (*ii*) o nome do anfitrião do recurso, (*iii*) o acrónimo da ontologia (p. ex., EFO), e (*iv*) a identificação do termo na mesma. Tomando como exemplo um dos termos da ontologia *Experimental Factor Ontology* (EFO), a decomposição da sua nomenclatura de endereçamento pode ser consultada na Tabela 3-11.

Verificação no sistema de cache

Uma área de acesso rápido é mantida pelo componente, com o histórico de valores de especificidade calculados em anteriores procedimentos de avaliação. Esta é uma estrutura simples em memória, onde cada entrada é descrita por um par {Chave-Valor}, à semelhança de outras já descritas. A cada Chave corresponde um valor alfanumérico, que contem o endereçamento único de uma dada anotação.

A cada Valor corresponde o valor de especificidade calculado, representado na Tabela 3-12. Antes de uma chamada efetiva ao repositório de ontologias é verificada a existência do endereçamento em causa perante esta estrutura. Se já existir um valor de especificidade este é anotado na estrutura de resposta geral do motor, sem necessidade de despender mais recursos para o seu cálculo.

Tabela 3-12 - Exemplo da estrutura de cache do componente OWL.

Chave	Valor
http://www.ebi.ac.uk/efo/EFO_0000400	0.750
http://www.ebi.ac.uk/efo/EFO_0000195	1.000

Cálculo de especificidade no repositório de ontologias

A cada entrada no ficheiro de configuração de ontologias corresponde, necessariamente, uma instância de base de dados onde se encontra representada, num modelo relacional, a ontologia indicada no acrónimo. Uma vez encontrado este será utilizado na construção de uma ligação ao motor relacional de base de dados que suporta o esquema da ontologia correspondente.

Uma vez encontrado um valor de especificidade para a anotação, uma nova entrada é feita sobre o sistema de cache de resultados do componente. À chave da entrada corresponderá o endereçamento da anotação, ao valor o resultado da função anterior. Este é posteriormente salvaguardado na estrutura de resultados do motor, ilustrada na Figura 3-12.

Conclusão do procedimento de avaliação

Uma vez encontrado o valor de especificidade de todas as anotações da classe enviada, este componente envia o resultado ao componente *Calculus* através de uma mensagem do tipo *OWLOutgoing*, descrita na Tabela 3-5, notificando-o do final de execução de uma sub-rotina, ilustrado na Figura 3-13 (etapa 2.1 do procedimento de avaliação).

3.2.3.8 Componente *Log*

O componente Log tem como objetivo canalizar o registo de todas as ocorrências do motor de análise e avaliação. Estas podem ser de dois tipos: (*i*) informação de progresso e (*ii*) exceções de processamento.

Sempre que existe alguma alteração de estado do sistema, tenha chegado o momento de entrar ou sair de uma fase de processamento ou exista algum outro tipo de informação que seja pertinente sobre os procedimentos de análise e avaliação, uma mensagem é enviada a este componente para que possa ser entregue ao administrador do sistema. Qualquer um dos componentes do sistema, durante a execução das suas funções, envia mensagens através do componente *Blackboard*, de modo a garantir a verbosidade no funcionamento do motor. Estas são da categoria *Info* e são encapsuladas em mensagens *Blackbaord* do tipo *Ingoing*, descrita na Tabela 3-5. Cada uma recebe uma descrição, que por nomenclatura se inicia com o nome da classe do componente, de onde a mensagem é originária.

Sempre que ocorra uma exceção de execução do motor de análise e avaliação é enviada uma mensagem a este componente, de modo a que fique registado o tipo de exceção, em que componente ocorreu e em que circunstâncias. A mensagem poderá ser do tipo (*i*) *Error* ou do tipo (*ii*) *ErrorWithThrowable*. A diferença entre ambas diz respeito à associação da informação da exceção gerada pelo compilador ao corpo da mensagem. O registo das mensagens poderá ser feito, por configuração, para dois destinos: barra de comandos ou base de dados.

3.2.4 Camada de Dados

Na camada de dados encontra-se o repositório de ontologias utilizado pelo motor de análise e avaliação para cálculo da especificidade das anotações. A existência desta camada, e a forma particular como está implementada, permitiu estabelecer um paradigma diferente daquele implementado por estudos anteriores [16], pois define cada ontologia como um modelo relacional suportado por um motor de base de dados.

Esta implementação tornou-se crucial no cumprimento dos requisitos não-funcionais de performance e modificabilidade, não apenas pela forma mais simples de acesso que permite, pois não se torna necessário navegar através de uma ontologia remota com recurso a sucessivas chamadas de HTTP, através de ligações TCP/IP, a repositórios públicos, o que em ontologias muito grandes poderia potencialmente ser bastante demorado e diminuir drasticamente a performance, mas porque assenta num modelo que pode ser utilizado de modo mais eficaz e eficiente no cálculo de especificidade de conceitos ontológicos, pois aposta na identificação de relações entre estes, e que pode ser expandido à medida daquilo que são as necessidades da plataforma.

Estruturalmente esta camada encontra-se segmentada em modelos relacionais, onde cada um corresponde a uma ontologia diferente. Cada modelo possui identidades, relações e outros objetos completamente separados dos restantes. Isto permite que cada ontologia possa ser consultada de modo independente, sem que exista qualquer relação com outras ontologias. Este fator torna-se importante na medida em que, estruturalmente, uma ontologia pode ser constituída por conceitos de uma outra, i.e., um conceito definido numa ontologia pode ser utilizado na descrição de áreas de uma outra ontologia.

Como exemplo, consideremos o termo "*Metabolite Profiling"* definido na ontologia *Ontology for Biomedical Investigations*[8] (OBI). Este termo, para além de originalmente definido nesta estrutura de conhecimento, é utilizado na construção de outras estruturas. É utilizado nas ontologias Eagle-i Resource Ontology (ERO), *OBI web service, development version* (webService) e *VIVO-ISF* (ISF). Isto significa que em ontologias diferentes este termo tem um conjunto de ascendentes e descendentes distintos. Caso existisse apenas um modelo relacional na plataforma onde todas as ontologias necessárias estivessem colocadas, em particular aquelas indicadas anteriormente, este termo teria como ascendentes e descendentes todos os termos que se apresentassem como tal em todas estas ontologias. O resultado do cálculo de especificidade seria assim baseado em múltiplas ontologias e estaria fora do âmbito desta tese. Através da segmentação de ontologias o resultado é limpo de relações não originais e apenas diz respeito à estrutura de conhecimento original onde o termo foi definido.

Tabela 3-13 - Lista de ontologias convertidas para o modelo racional.

Nome	**Acrónimo**	**Descrição**
BioAssay Ontology	BAO	Uma descrição semântica de bioensaios e os resultados de rastreio de alto rendimento
BRENDA Tissue and Enzyme Source Ontology	BTO	Um vocabulário estruturado e controlado para a fonte de uma enzima
Chemical Methods Ontology	CHMO	Uma ontologia de métodos químicos
Clinical Measurement Ontology	CMO	Uma ontologia para padronizar registros de medições morfológicas e fisiológicas
Experimental Factor Ontology	EFO	Uma ontologia de modelagem de variáveis experimentais usadas em vários recursos na EBI
Environment Ontology	ENVO	Ontologia de caraterísticas ambientais e habitas
Gene Ontology	GO	Uma estrutura para a modelação da biologia

[8] https://bioportal.bioontology.org/ontologies/OBI

Nome	Acrónimo	Descrição
Health Level Seven	HL7	Ontologia para descrição formal de conceitos de segurança e privacidade na área de *Healthcare Information Technology*
Mammalian Phenotype Ontology	MP	Ontologia para fornecer termos-padrão de anotação de dados fenotípicos de mamíferos
Mass Spectrometry Ontology	MS	Um vocabulário estruturado e controlado para a anotação de experiências na área de espectrometria de massa
Neural ElectroMagnetic Ontology	NEMO	Ontologia que descreve classes de potenciais cerebrais relacionados a eventos (ERP) e suas propriedades
Ontology for Biomedical Investigations	OBI	Uma ontologia que descreve protocolos, instrumentação, materiais usados, dados gerados e tipos de análise em investigações biomédicas
Phenotypic Quality Ontology	PATO	Ontologia utilizada em conjunção com outras na descrição de fenótipos
Plant Ontology	PO	Um vocabulário controlado que descreve a anatomia vegetal e morfologia e estágios de desenvolvimento para todas as plantas
	SDON	
Systems Biology Ontology	SBO	Um conjunto de vocabulários relacionais controlados de termos geralmente usados em Biologia de Sistemas, particularmente em modelação computacional
Units of Measurement Ontology	UO	Ontologia que descreve unidades métricas para utilização com a ontologia PATO
Experimental Conditions Ontology	XCO	Ontologia que representa as condições sobre as quais são feitas medições fisiológicas e morfológicas

Neste processo de segmentação ontológico foi necessária a transposição das ontologias, geralmente em formato OWL ou TTL, descritos na Secção 2.1, para um modelo relacional. Esta transposição foi feita com recurso a uma ferramenta de conversão desenvolvida na FCUL, a OWLtoSQL, detalhada na Secção 2.5.1. Foram inicialmente convertidas dezoito ontologias, descritas na Tabela 3-13. Estas ontologias foram aquelas que no espectro do caso de um estudo anterior [16] maior utilização tiveram.

Para cada uma das ontologias da solução, e para além das entidades, respetivas propriedades e relações, acresce um conjunto de ferramentas internas ao modelo, desenvolvidas em SQL que possibilitam o cálculo da especificidade de uma dada

anotação para um termo na ontologia. Estas colaboram entre si de modo a que o algoritmo de cálculo de especificidade de uma anotação seja executado. Estas ferramentas são:

a. **Procedimento *f_get_owlid_from_iri***: Método que dado o URI de uma anotação, com referência a um termo ontológico, retorna o seu identificador único, no modelo relacional.

b. **Procedimento *f_concept_ancestors_count***: Método que dado o identificador único de uma anotação retorna o número de termos ascendentes, no modelo.

c. **Procedimento *sp_conceptspec***: Método que dado o URI de uma anotação retorna o valor de especificidade do termo ontológico, ao qual se refere.

Procedimento f_get_owlid_from_iri

Através deste procedimento é possível obter o identificador único da anotação em avaliação, através do endereçamento único URI (variável *concept_iri*) do termo ontológico com o qual estabelece uma ligação. Com este valor é possível questionar a entidade do modelo, descrito na Secção 3.3.1, que regista todos os objetos existentes na ontologia à qual o modelo se refere. A sua implementação em SQL junto do modelo relacional de cada ontologia foi a seguinte:

```
BEGIN
   DECLARE result_value   INTEGER DEFAULT 0;
   SELECT o.id
     INTO result_value
     FROM owltosql_bao.owl_objects o
    WHERE o.type = 'Class' AND LOWER(o.iri) = LOWER(concept_iri);
   RETURN result_value;
END
```

Este procedimento procura identificar instâncias cujo tipo de objeto na ontologia se restrinja a classes (*o.type = 'Class'*) e o endereçamento único seja aquele que foi passado como parâmetro do procedimento (*LOWER(o.iri) = LOWER(concept_iri)*).

Procedimento f_concept_ancestors_count

Estre procedimento identifica o número de termos ascendentes do termo ontológico identificado pela anotação em avaliação, através do seu valor único (variável *owl_obj_id*), no modelo. De acordo com este, são consideradas termos ascendentes aqueles que possuam uma relação de hierarquia, com o termo dado, sendo identificadas como superclasse da subclasse fornecida. A sua implementação em notação SQL junto do modelo relacional de cada ontologia é a seguinte:

```
BEGIN
   DECLARE result_value   INTEGER DEFAULT 0;
     -- get all ancestors count for the given concept
     SELECT count(h.superclass) hopcount
       INTO result_value
       FROM hierarchy h
            INNER JOIN owl_objects o ON h.superclass = o.id
            INNER JOIN names n ON h.superclass = n.id
      WHERE     h.subclass = owl_obj_id
            AND h.superclass <> owl_obj_id
            AND o.type = 'Class'
   ORDER BY h.distance DESC, h.superclass;
   RETURN (result_value);
END
```

Este procura contabilizar instâncias que sejam superclasses (*count(h.superclass)*) da subclasse (*h.subclass = owl_obj_id*) fornecida, excluindo-a da lista de resultados (*h.superclass <> owl_obj_id*).

Procedimento sp_conceptspec

Este procedimento tem como objetivo calcular o valor de especificidade da anotação em avaliação através da indicação do endereçamento único, o URI, do termo ontológico a que se refere. Utiliza os procedimentos anteriores de modo a cumprir com o seu objetivo. A sua implementação em SQL pode ser consultada no Anexo N. O algoritmo de cálculo pode ser explicado pela seguinte notação de pseudo-código:

```
CALL f_get_owlid_from_iri WITH (concept_iri)
SET class identification TO the returned value
IF class identification is greater than 0
       CALL f_concept_ancestors_count with (class identification)
       SET ancestor count TO the returned value
       IF ancestor count is greater than 0
              FIND all class's leaf descendants
              IF class's leaf descendants is greater than 0
                     WHILE leaf descendants counter is greater than 0
                            CALL f_concept_ancestors_count with (leaf
                            identification)
                            SET leaf ancestor's count TO the returned
                     value
                            COMPUTE difference between leaf and class
                            distance
                     SET specification value to class's ancestor count
                     divided by class's ancestor count plus the class's
                     leaf descendant's average distance
              ELSE
                     SET specification value TO 1
       ELSE
              SET specification value TO 0
ELSE
       SET specification value TO -1
RETURN specification
```

A implementação deste algoritmo tem como objetivo auxiliar na determinação da qualidade semântica de uma anotação, como indicado na Secção 3.1. Assim começa por determinar o número de termos ascendentes, a partir do termo ontológico indicado, com o auxílio do procedimento *f_concept_ancestors_count*. De seguida obtém uma lista de todos os seus descendentes folha, i.e., termos que façam parte da ramificação presente na árvore da ontologia a partir do termo e que não tenho por si próprios mais descendentes. Por cada um destes termos folha é determinada a distância entre estes e o termo para o qual se pretende calcular a especificidade. Por fim calcula esse valor tendo em conta o total de termos ascendentes e a média de distâncias encontradas nos seus termos folha.

Para termos sem ascendentes o valor de especificidade é 0, o que denota um termo sem um grande conhecimento semântico associado. Para termos sem descendentes o valor de especificidade é 1, o que denota um termo para o qual já não é possível aumentar o conhecimento semântico proporcionado. Se o procedimento *f_get_owlid_from_iri* retorna um valor igual a 0 significa que o termo ontológico não existe no repositório, obtendo por isso um valor de especificidade de -1.

3.3 Implementação da arquitetura

Nesta secção são descritos os detalhes técnicos de implementação das várias camadas que constituem a solução de contribuição desta tese. Descreve a implementação das camadas de fontes de dados, do motor de análise e avaliação (MAA), da interface computacional e da interface de utilizador.

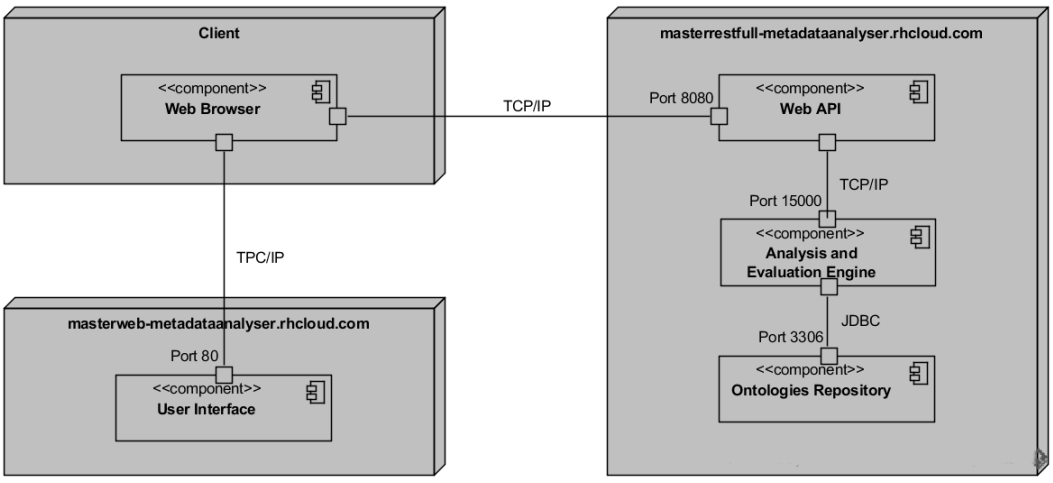


Figura 3-15 - Diagrama de implementação da solução da tese.

Cada uma das camadas descritas na Secção 3.2 foi traduzida em um componente físico, implementado, como ilustrado no diagrama de implementação da Figura 3-15, sobre um dos nós aplicacionais da plataforma *Cloud* da *Red Hat*, descrita na Secção 2.5.4. Por limitações de ordem técnica por parte da plataforma, os componentes de repositório

de ontologias, do motor de análise e avaliação e interface computacional foram colocados num único nó. Este apenas permite comunicação com exterior através da porta TCP 8080, i.e., apesar de existir a hipótese de cada um dos componentes operar em um nó diferente, cumprindo com os requisitos não-funcionais de performance e modificabilidade, descritos na Tabela 3-3, todos eles, à exceção da interface de utilizador, foram instalados no mesmo nó aplicacional, com uma única porta de saída ligada à interface computacional.

3.3.1 Fonte de dados

O repositório de ontologias foi implementado sobre o motor de base de dados relacionais *MySQL* versão 5.5.45, descrito na Secção 2.5.4, hospedado no nó aplicacional da *Red Hat*, ilustrado no diagrama da Figura 3-15, com o IP 127.3.70.2. Este repositório é constituído por um conjunto de base de dados distintas. Cada uma corresponde à representação relacional de uma ontologia em particular. Esta implementação foi realizada com auxílio à ferramenta de conversão OWLtoSQL, descrita na Secção 2.5.1, englobada em um projeto *Java* desenvolvido com auxílio ao IDE *Eclipse*.

Cada uma das ontologias, enumeradas na Tabela 3-13, foi obtida a partir do seu repositório público e salvaguardada localmente em formato *Web Ontology Language* (OWL) ou *Terse RDF Triple Language* (TTL), descritos na Secção 2.2. A ferramenta de conversão, OWLtoSQL, foi posteriormente executada por cada uma das ontologias, sendo o motor de base de dados anterior o alvo do processo de conversão.

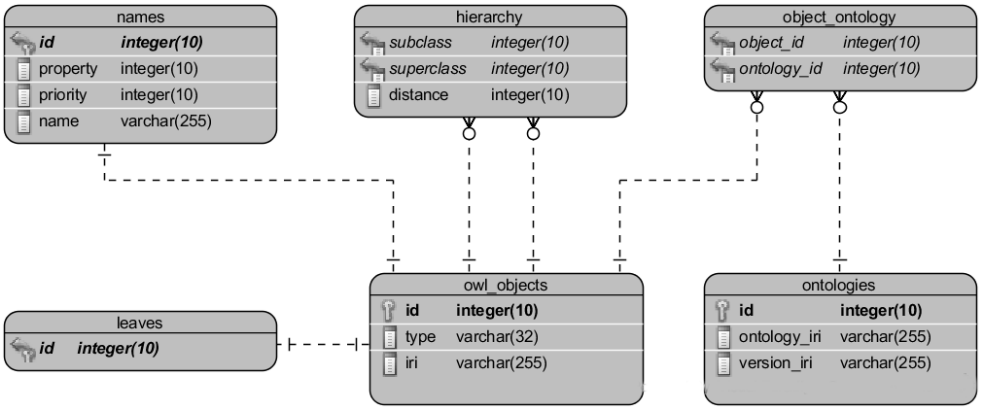


Figura 3-16 - Diagrama de Entidade-Relação do repositório de ontologias.

Durante o processo de conversão foi gerado um modelo relacional para cada uma das ontologias, descrito no diagrama de entidade-relação da Figura 3-16, onde a sua informação foi enquadrada. O diagrama é composto por seis entidades: (*i*) *owl_objects*, (*ii*) *ontologies*, (*iii*) *object_ontology*, (*iv*) *hierarchy*, (*v*) *names* e (*vi*) *leaves*.

A primeira entidade, *owl_objects*, regista todos os objetos encontrados na ontologia, como classes ou propriedades. A segunda, *ontologies*, indica quais as ontologias presentes no modelo, pois podem ser mais do que uma. A terceira, *object_ontology*, relaciona cada objeto com a ontologia onde este foi encontrado. A quarta, *hierarchy*, estabelece uma relação de hierarquia entre os vários objetos encontrados, através da análise dos triplos de RDF [22], como *subClassOf*. A quinta, *names*, contém os nomes de cada conceito das ontologias. A sexta entidade, *leaves*, enumera todos os objetos que não tenham descendentes, do ponto de vista da hierarquia implementada.

Acresce a cada modelo um conjunto de procedimentos, desenvolvidos em SQL: (*i*) *f_get_owlid_from_iri*, (*ii*) *f_concept_ancestors_count* e (*iii*) *sp_conceptspec*, descritos na Secção 3.2.4, que recorrem a todas as referidas entidades no processamento dos seus objetivos. Os dois primeiros implementados como objetos *Function*, o último como objeto *Stored Procedure*.

Tabela 3-14 - Lista de bases de dados presente no repositório de ontologias.

Ontologia	**Base de dados**
BioAssay Ontology	owltosql_bao
BRENDA Tissue and Enzyme Source Ontology	owltosql_bto
Chemical Methods Ontology	owltosql_chmo
Clinical Measurement Ontology	owltosql_cmo
Experimental Factor Ontology	owltosql_efo
Environment Ontology	owltosql_envo
Gene Ontology	owltosql_go
Health Level Seven	owltosql_hl7
Mammalian Phenotype Ontology	owltosql_mp
Mass Spectrometry Ontology	owltosql_ms
Neural ElectroMagnetic Ontology	owltosql_nemo
Ontology for Biomedical Investigations	owltosql_obi
Phenotypic Quality Ontology	owltosql_pato
Plant Ontology	owltosql_po
	owltosql_sdon
Systems Biology Ontology	owltosql_sbo
Units of Measurement Ontology	owltosql_uo
Experimental Conditions Ontology	owltosql_xco

O nome de cada uma das bases de dados é composto pelo prefixo *owltosql_*, o nome da ferramenta de conversão, seguido do seu acrónimo, detalhado na Tabela 3-14. Cada uma fica disponível para consulta através de uma ligação ao motor relacional na porta 3306, onde é indicado o nome respetivo. A linguagem de consulta é o SQL.

3.3.2 Motor de análise e avaliação

O motor de análise e avaliação (MAA) foi desenvolvido como um projeto *Java* com auxílio do IDE *Eclipse*, à semelhança da interface computacional, e exportado como um arquivo Java executável (JAR). Como controlador de dependências do seu desenvolvimento foi usado o *Maven*. Este foi colocado em execução no mesmo nó aplicacional da fonte de dados, como ilustrado no diagrama de implementação da Figura 3-15.

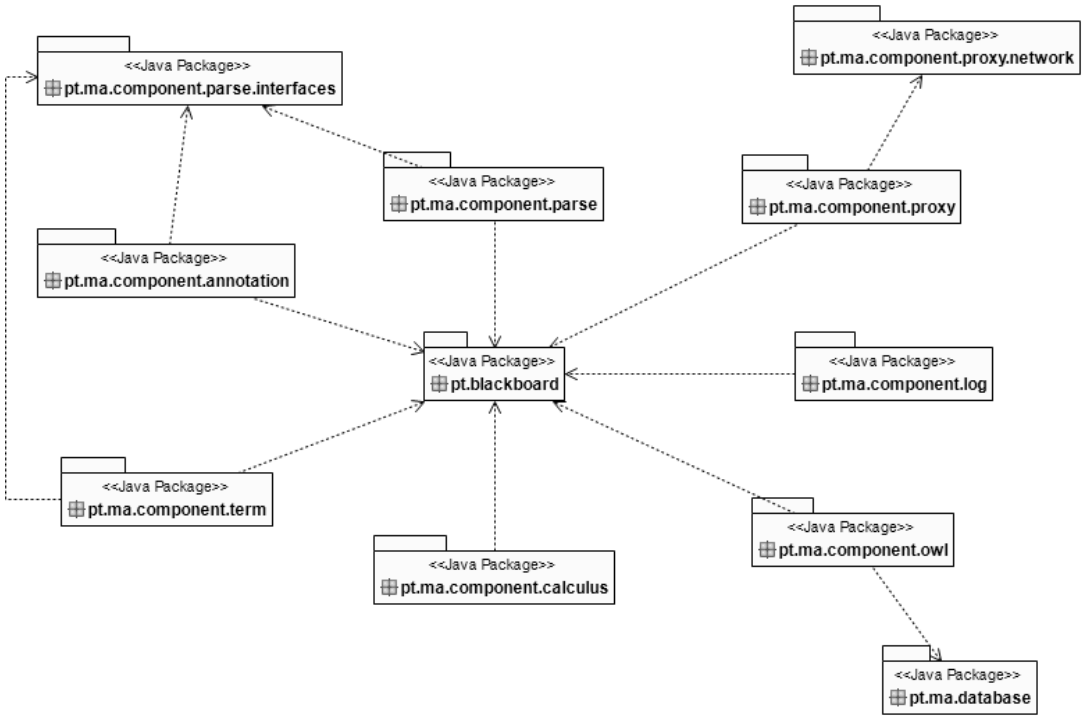


Figura 3-17 - Diagrama de dependência de pacotes do MAA.

Este projeto foi implementado através de vários pacotes Java distintos, que organizam logicamente todos as classes utilizadas para garantir a funcionalidade de vários componentes. Dos vários pacotes existentes, aqueles que constituem a base do motor estão descritos no diagrama de pacotes ilustrado na Figura 3-17. Neste conjunto estão compreendidos os pacotes que suportam os componentes descritos na arquitetura do motor de análise e avaliação, ilustrada na Figura 3-15. Este compreende os pacotes: (*i*) *pt.ma.component.proxy*, (*ii*) *pt.ma.component.parser*, (*iii*) *pt.ma.component.annotations*, (*iv*) *pt.ma.component.term*, (*v*) *pt.ma.component.calculus*, (*vi*) *pt.ma.component.owl*, (*vii*) *pt.ma.component.log* e (*viii*) *pt.ma.component.blackboard*. Acrescem os pacotes (*ix*)

pt.ma.component.proxy.network, (*x*) *pt.ma.component.parser.interfaces* e (*xi*) *pt.ma.database*, como segunda linha de pacotes essenciais à funcionalidade do motor. O código fonte de todo o projeto pode ser consultado através do endereço: http://*gitlab.com/inacio.bruno/metadataanalyser-master*.

Pacote *blackboard*

O pacote *pt.blackboard* implementa um conjunto de classes *Java* que suportam a funcionalidade do componente *Blackboard*, ilustrado no diagrama de componentes da Figura 3-6, como meio de transmissão de mensagens entre componentes que constituem este motor. O diagrama de todas as classes deste pacote pode ser consultado no Anexo A. Uma instância da classe *Blackboard* é a primeira a ser criada no arranque do motor. Esta é criada através da interface *IBlackboard*, de modo a implementar o paradigma de polimorfismo que o Java possibilita.

A sua funcionalidade é garantida, sobretudo, pela inclusão no projeto da biblioteca *Java TupleSpaces* [77], que implementa um repositório de Tuplos, unidade {Chave-Valor}, onde cada um representa uma ação, ou mensagem, que foi enviada de um componente para outro, e que pode ser acedida em modo concorrencial, de acordo com a Secção 3.2.3.1.

Pacotes *proxy* e *proxy.network*

O pacote *pt.ma.component.proxy* compreende um conjunto de classes *Java* que suportam a funcionalidade do componente *Proxy*, ilustrado no diagrama de componentes da Figura 3-6, como plataforma de intercâmbio de mensagens entre os clientes do motor e os seus componentes internos. O seu diagrama de classes pode ser consultado no Anexo B. Uma instância da classe *ProxyObject* é a última a ser criada no arranque do motor de análise e avaliação, de modo a evitar que sejam adicionados pedidos externos ao *Blackboard* antes de o motor de análise e avaliação estar completamente inicializado. Esta recebe a instância da interface *IBlackboard*, do pacote *ma.blackboard*, no seu construtor, através do padrão *Dependency Injection*.

Recorre ao pacote *pt.ma.component.proxy.network*, descrito no diagrama de classes do Anexo F, de modo a implementar uma camada de ligação de rede, necessária à troca de mensagens do motor com os seus clientes. Através da instância da classe *Interface* é possível manter uma escuta continua sobre a porta TCP 15000, de forma a receber mensagens dos clientes externos, como ilustrado no diagrama de implementação da Figura 3-15. O padrão *Observer*, descrito em [76], foi utilizado sobre esta classe, através da extensão da classe *Observable* do pacote Java *java.util*, de forma a garantir que as mensagens recebidas possam ser processadas. Através desta instância é possível também enviar mensagens a clientes, informando sobre o progresso do pedido ou entregando o

resultado do mesmo. A sua funcionalidade assenta sobretudo na implementação da biblioteca Java *java.net*, em particular das classes *ServerSocket* e *Socket* do conceito *Java Sockets*, detalhado na Secção 2.5.2.

Este método de comunicação entre processos foi escolhido entre outros, p. ex., o *Java Remote Method Invocation* (RMI), pela sua simplicidade de implementação, pois apenas se torna necessária a inclusão e implementação da biblioteca de *Java* referida.

Pacotes *parser, annotations, terms* e *parser.interfaces*

Os pacotes *pt.ma.component.parser*, *pt.ma.component.annotations* e *pt.ma.component.terms* englobam um conjunto de classes *Java* que suportam a funcionalidade dos componentes *Parser*, *Annotations* e *Terms*, da arquitetura ilustrada na Figura 3-15, na execução do procedimento de análise de um ficheiro de metadados, detalhado na Secção 3.2.3.3. Os seus diagramas de classes podem ser consultados no Anexo C (*Parser*) e Anexo D (*Annotations* e *Terms*). Durante o arranque do motor é criada em primeiro lugar uma instância da classe *Java TermObject*, seguida de uma instância da classe Java *AnnotationObject* e por último, uma instância da classe Java *ParserObject*. Todas elas recebem uma instância da interface *IBlackboard*, do pacote *ma.blackboard*, através do padrão *Dependency Injection*.

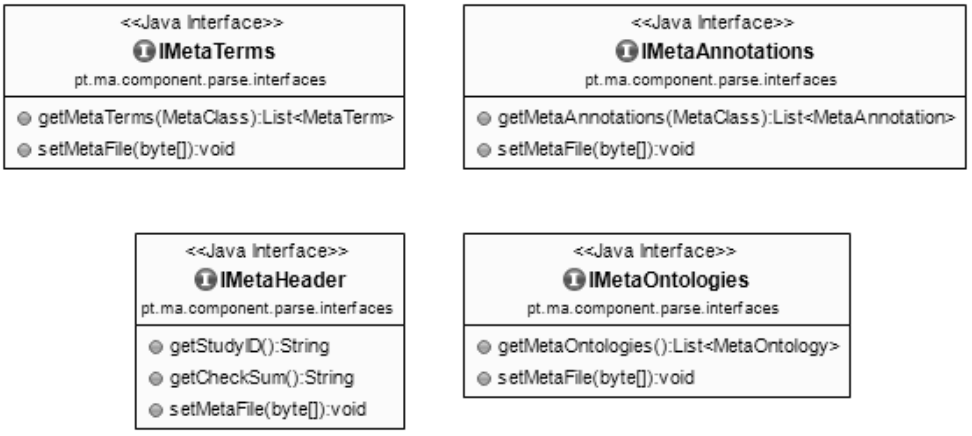


Figura 3-18 - Diagrama de classes do pacote ma.component.parser.interfaces.

Todos estes pacotes recorrem a um conjunto de interfaces contidos no pacote *pt.ma.component.parser.interfaces*, ilustrados no diagrama de classes da Figura 3-18, cuja funcionalidade é descrita na Tabela 3-9, de modo a instanciar as classes que possibilitam a análise de um dado ficheiro de metadados de acordo com o tipo de repositório indicado. Estes descrevem as funções gerais a considerar na análise de um ficheiro de metadados, deixando para a sua implementação os detalhes concretos e distintos a cada repositório de metadados. Este conjunto é de extrema importância para a concretização de um dos requisitos não-funcionais, a modificabilidade, descrito na Tabela

3-3, pois permite que a solução se adapte potencialmente a qualquer repositório de dados científicos.

Pacotes *calculus, owl* e *database*

Os pacotes *pt.ma.component.calculus* e *pt.ma.component.owl* englobam um conjunto de classes Java que suportam a funcionalidade dos componentes *Calculus* e *OWL*, da arquitetura do motor de análise e avaliação ilustrada na Figura 3-6, na execução do procedimento de avaliação das anotações de um ficheiro de metadados, detalhado na Secção 3.2.3.6. Os seus diagramas de classes podem ser consultados no Anexo E. Durante o arranque do motor são criadas instâncias da classe *Java CalculusObject* e da classe *OWLObject*. Ambas recebem uma instância da interface *IBlackboard*, do pacote *ma.blackboard*, cujo diagrama de classes pode ser consultado no Anexo A, através do padrão *Dependency Injection*.

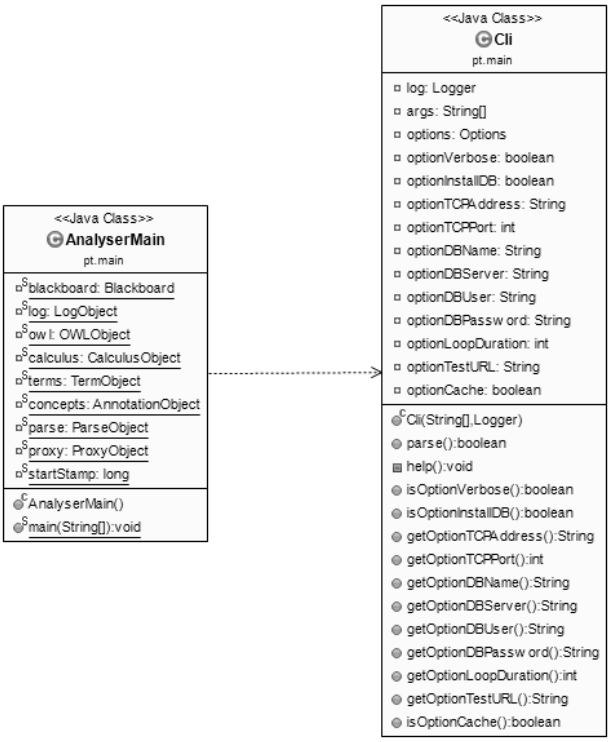


Figura 3-19 - Diagrama de classes do pacote pt.ma.main.

O pacote *pt.ma.database*, ilustrado no diagrama de pacotes da Figura 3-17, é utilizado para estabelecer a camada de abstração na comunicação ao repositório de ontologias, por parte do componente OWL. Através deste pacote são feitas as ligações *Java Database Connectivity* (JDBC, porta TCP 3306), ilustradas no diagrama de implementação da Figura 3-15, ao repositório de ontologias suportado por *MySQL*.

Pacote Log

O pacote *pt.ma.component.log* engloba um conjunto de classes Java que suportam a funcionalidade do componente Log, da arquitetura ilustrada no diagrama da Figura 3-15, de registo de informação de progresso e ocorrência de exceções na execução do motor de análise e avaliação. O seu diagrama pode ser consultado no Anexo G.

Parametrização de arranque do motor

A ordem de arranque dos componentes do motor de análise e avaliação é feita da seguinte forma:

1. Componente *Blackboard*;
2. Componente *Log*
3. Componente *OWL*
4. Componente *Calculus*
5. Componente *Terms*
6. Componente *Annotations*
7. Componente *Parser*
8. Componente *Proxy*

Para além da ordem de arranque do motor, todo este procedimento é parametrizável por um conjunto de valores que vão necessariamente influenciar a execução e resultado do motor. O procedimento de arranque está implementado sobre as classes que fazem parte do pacote Java *pt.ma.main*, ilustrada no diagrama de classes da Figura 3-19, em particular sobre a classe *AnalyserMain*. Sempre que o ficheiro JAR do motor é executado é feita uma chamada ao procedimento *Main* desta classe, iniciando assim o arranque do motor.

Tabela 3-15 - Lista de parametros de arranque do motor de análise e avaliação.

Flag	Nome	Descrição
-h	Ajuda	Mostra o ecrã de auxílio
-v	Verbosidade	{*true*|*false*} Controla a visibilidade da informação
-i	InstalaDB	{*true*|*false*} Controla a instalação de raiz da base de dados
-a	EndereçoTCP	Endereço TCP/IP de escuta para o Socket de entrada
-p	PortaTCP	Porta TCP/IP de escuta para o Socket de entrada
-n	NomeDB	Nome da base de dados de referência no repositório
-s	ServidorDB	Endereço TCP/IP do servidor do repositório
-u	UtilizadorDB	Nome de utilizador do repositório
-w	ChaveDB	Palavra-chave do utilizador do repositório

Flag	Nome	Descrição
-l	DuraçãoLoop	{inteiro} Valor de milissegundos para controlo de *loop*
-c	Cache	{true|false} Controla a utilização de cache pelo OWL

Durante a execução do motor ficam guardadas nesta classe as instâncias de todos os componentes do motor, e recai sobre ela a responsabilidade de ler todos os parâmetros de arranque. Estes são passados na sintaxe de execução do ficheiro JAR do motor. A sintaxe de execução é a seguinte:

- *java -jar {nome do ficheiro}.jar -v {verbosidade} -i {instala base de dados} -a {endereço do motor} -p {porta TCP} -n {base de dados} -s {endereço BD} -u {utilizador BD} -w {palavra-chave BD}*

A recolha e salvaguarda, através das suas propriedades, dos parâmetros é feita pela classe *Cli*, ilustrada no diagrama de classes da Figura 3-19, que através da implementação da biblioteca *org.apache.commons.cli*, que apresenta um conjunto de funcionalidades para tratar de parâmetros submetidos por barra de comandos, estabelece uma estrutura de gestão de parâmetros. A lista de parâmetros recebidos pela aplicação pode ser consultada na Tabela 3-15.

3.3.3 Interface Computacional

A interface computacional foi desenvolvido como um projeto Java, com auxílio do IDE Eclipse, e exportado como um arquivo *Java Web* (WAR), hospedado em um dos nós aplicacionais da plataforma *Cloud* da *Red Hat*, com o endereço público:

- *http://masterrestfull-metadataanalyser.rhcloud.com:8080*

Como controlador de dependências de bibliotecas foi usado novamente o *Maven*, um controlador particular para o *Java*. O seu código fonte pode ser consultado através do endereço: *https://gitlab.com/inacio.bruno/metadataanalyser-webapi*. Este interface foi colocado em execução no mesmo nó aplicacional do motor de análise e avaliação e da fonte de dados, ilustrado no diagrama de implementação da solução na Figura 3-15. O servidor aplicacional escolhido para suportar a interface foi o *Apache Tomcat*, versão 7, descrito na Secção 2.5.4.

O paradigma escolhido para a implementação deste interface foi o *RESTFul*, de *Representational State Transfer* (REST), descrito na Secção 2.5.4, onde dados e funcionalidade são considerados como recursos e acedidos por HTTP, através da utilização de URI, i.e. endereços únicos na *Internet*. Através da utilização deste paradigma foi possível definir um conjunto de recursos de modo a garantir as funcionalidades de um interface público, desenvolvido com o intuito de fornecer uma

camada de abstração à utilização do motor de análise e avaliação, que pudesse ser acedida através de um conjunto de operações uniforme, unicamente identificadas e auto descritas.

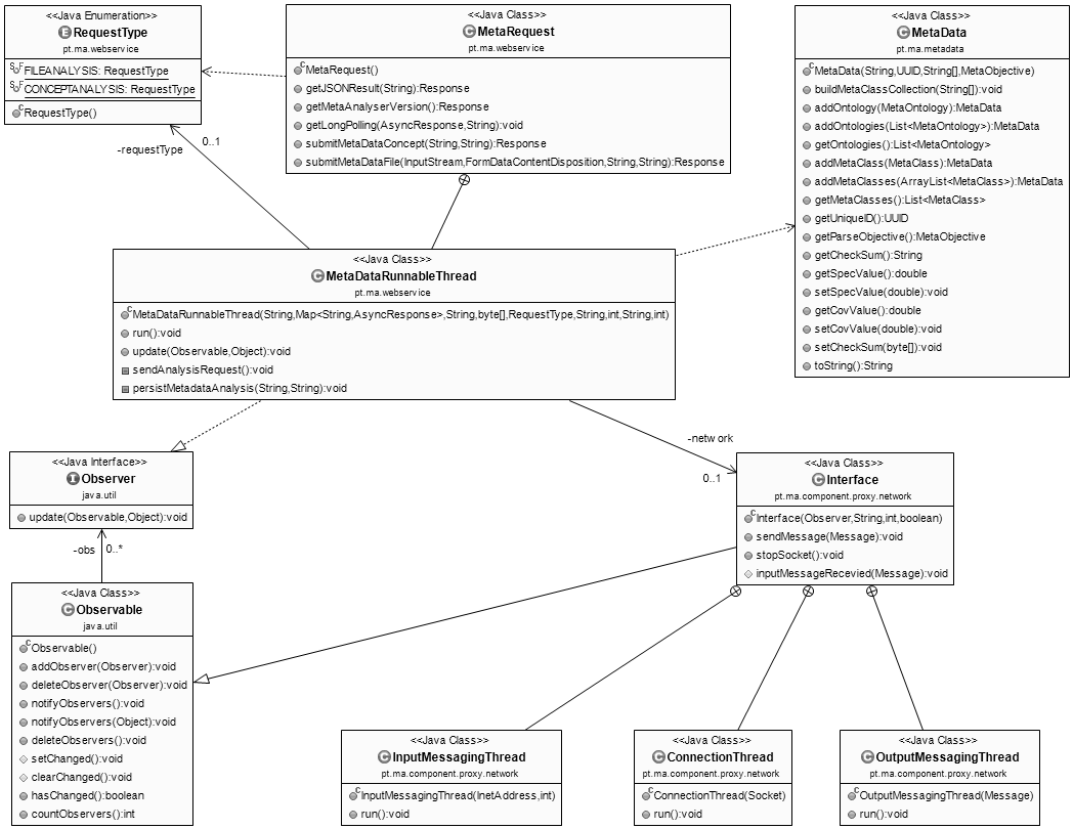


Figura 3-20 - Diagrama de classes do interface computacional da plataforma.

Para a implementação deste foi utilizada a biblioteca Java para *RESTful Web Services* (JAX-RS), através da inclusão da plataforma *Jersey*, descrita na Secção 2.5.2. O projeto foi organizado em um conjunto de pacotes Java, que englobam um conjunto de classes, as quais suportam toda a funcionalidade da Camada *Web* descrita na Secção 3.2.2. O mais relevante destes pacotes (*pt.ma.webservice*) encontra-se descrito no diagrama de classes ilustrado na Figura 3-20.

Tabela 3-16 - Mapeamento entre recursos da camada computacional.

Ação	**Método**	**Operação HTTP**
/submitfile	submitMetaDataFile()	PUT
/polling/{poolid}	getLongPolling()	GET
/tojson	getJSONResult()	GET
/version	getMetaAnalyserVersion()	GET

Classes *MetaRequest* e *MetaDataRunnableThread*

A classe *MetaRequest*, ilustrada no diagrama de classes da Figura 3-20, implementa todos os métodos que suportam o mapeamento com o URI das ações públicas disponíveis pela interface computacional, i.e., sempre que o cliente invoca um dos recursos disponibilizados pelo serviço, através de um pedido HTTP, um método da classe é executado. Na Tabela 3-16 encontram-se enumeradas as ações descritas na Secção 3.2.2e respetivo mapeamento interno aos métodos da classe *MetaRequest*.

Por cada ficheiro de metadados enviado, quer através da sua localização na *Internet* (URL), quer através do formato binário, é invocado o método *submitMetaDataFile()*. Por cada chamada é criada uma instância da classe *MetaDataRunnable Thread*, ilustrada no diagrama da Figura 3-20, que através da implementação da interface *Runnable*, da biblioteca *Java java.lang*, estabelece as condições necessárias para execução paralela deste pedido como um *Java Thread*, desde o seu envio ao motor de análise e avaliação até à conclusão do processo por parte deste. Cada um dos pedidos recebidos é colocado numa coleção de *Threads Java* (*Thread Pool*) em execução, até que todas as etapas do processo de análise e avaliação estejam completas. Esta coleção é mantida em memória através da classe *ExecutorService*, da biblioteca Java *java.util.concurrent*.

A classe *MetaDataRunnableThread*, à semelhança da implementação do componente *Proxy* do motor de análise e avaliação, recorre ao pacote *pt.ma.component.proxy.network*, ilustrado no diagrama de pacotes da Figura 3-17, de modo a implementar uma camada de ligação de rede necessária à troca de mensagens com o motor de análise e avaliação. Através da instância da classe *Interface* é possível manter uma escuta contínua sobre uma porta TCP dinâmica, de modo a receber mensagens, como ilustrado no diagrama de implementação da Figura 3-15. O padrão *Observer*, descrito em [76], foi implementado sobre esta classe, através da extensão da classe *Observable* do pacote Java *java.util*, de forma a garantir que as mensagens recebidas possam ser processadas. A sua funcionalidade assenta sobretudo na implementação da biblioteca Java *java.net*, em particular das classes *ServerSocket* e *Socket*, do conceito *Java Sockets*, descrito na Secção 2.5.2.

Procedimento de *Long Polling*

O período de processamento de um pedido, ao motor de análise e avaliação, poderá ser longo, ou maior do que aquele suscetível de ser aceite por uma chamada HTTP, ou maior do que aquele que o utilizador considere razoável esperar. Para resolver estas questões foi implementado o conceito de *Long Polling*, descrito na Secção 2.5.3 e ilustrado na Figura 3-21, que simula o envio de informação automática por parte do servidor aplicacional ao cliente.

Para tal, foi definido o método *getLongPolling()* na classe *MetaRequest*, enumerado na Tabela 3-16, que recebe como parâmetro um identificador único de pedido (*poolid*), gerado e enviado ao cliente na primeira chamada ao método *submitMetaDataFile()* da mesma classe, e uma instância do objeto *AsyncResponse*, uma propriedade da biblioteca JAX-RS, que representa a resposta a ser dada ao cliente e que permite o tratamento de

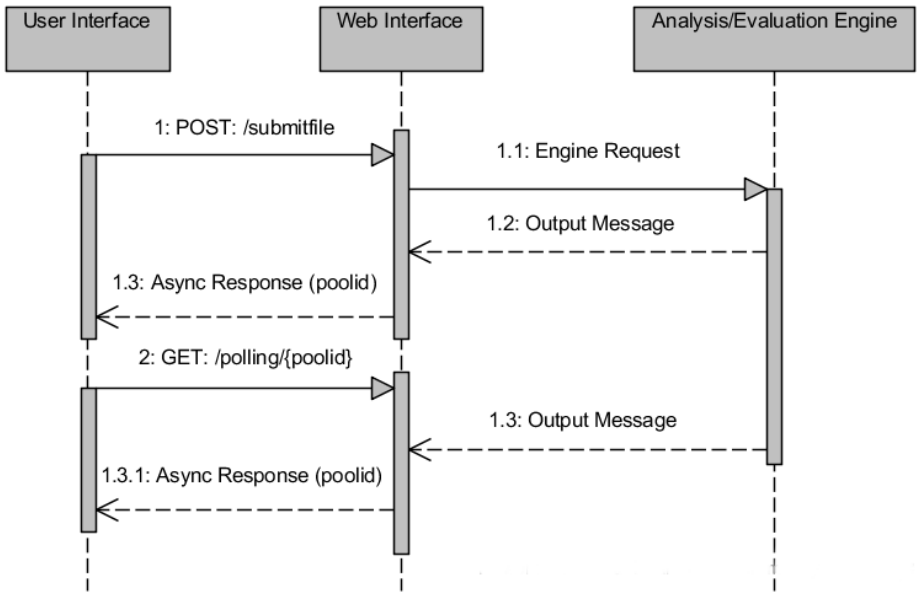


Figura 3-21 - Diagrama de iteração do conceito Long Pooling.

chamadas em modo assíncrono por parte do servidor através da suspensão da sua execução até um determinado momento, i.e., uma vez enviado um pedido pelo cliente, computacional ou humano, este não necessita de ser respondido de imediato, de modo síncrono, como normalmente acontece num pedido por HTTP.

O identificador único (*poolid*) e o objeto de resposta são salvaguardados em memória por uma estrutura {Chave-Valor}, à semelhança de outras já referidas nesta tese. Sempre que chega uma mensagem por parte do motor de análise e avaliação de progresso ou finalização do processo, processada pelo *Java Thread* correspondente ao pedido inicial, é lido o identificador único da mesma e obtido o objeto *AsyncResponse* que lhe corresponde, na referida estrutura. Uma vez processada, a resposta ao cliente pode ser resumida com a mensagem desejada (etapas 1.3 e 1.3.1 da Figura 3-21). Este modelo de reposta foi inspirado no modelo descrito em [42], para o formato ISA-TAB.

Resultado final

O resultado final do processo de análise e avaliação é reconvertido a partir da mensagem enviada pelo motor, descrita na Tabela 3-6, e convertido em uma representação JSON, notação descrita na Secção 2.5.5, do seguinte modo:

```
{
  "id": "MTBLS1",
  "uniqueID": "d70a96e5-6662-4279-a129-7ea656d8fcbd",
  "checkSum": "d87173d6227f95bfeb2358e9d4d9087d93943a16",
  "parseDate": 1460841776506,
  "parseDuration":46084,
  "specValue": 0.75,
  "covValue": 1.0
  "ontologies": [{
     "id": "ONTO_0",
     "uri": "http://data.bioontology.org/ontologies/EFO"
   }],
  "metaClasses": [{
     "id": "Design_1",
     "uniqueID": "a3bdf7d5-b81c-4535-badb-aeeee8655ddc",
     "name": "Design",
     "specValue": 0.75,
     "covValue": 1.0,
     "metaAnnotations": [{
        "id": "ANNO_0",
        "uniqueID": "88957c10-6bbf-4a6e-8930-ee1efb5d93c5",
        "uri": "http://www.ebi.ac.uk/efo/EFO_0000400",
        "specValue": 0.75
      }],
     "metaTerms": [{
        "id": "TERM_0",
        "uniqueId": "e23b45f5-e545-4e9a-9115-016e150563aa",
        "name": "diabetes mellitus"
      }]
   }],
  "parseObjective": "METADATAANALYSIS"
}
```


O resultado do motor é uma instância da estrutura de classes *Java* ilustrada no diagrama de classes da Figura 3-12, convertida para a notação JSON através da biblioteca *Gson*, uma biblioteca *Java* para conversão de objetos de e para representações JSON.

3.3.4 Interface de Utilização

A interface de utilização da solução foi desenvolvido como um projeto PHP, linguagem descrita na Secção 2.5.3, com auxílio do *IDE Eclipse*. Este foi implementado com o auxílio da plataforma *CakePHP*, também ela descrita na Secção 2.5.3, que permite o rápido desenvolvimento de projetos para a *Internet* utilizando a linguagem PHP. Encontra-se hospedado em um dos nós aplicacionais da *Cloud* da *Red Hat*, descrita na Secção 2.5.4, como ilustrado no diagrama de implementação da Figura 3-15, com o endereço:

- *http://masterweb-metadataanalyser.rhcloud.com*

De modo a suportar a funcionalidade deste interface foi programado um conjunto de funções *Javascript*, com o auxílio da plataforma *jQuery*, descrita na Secção 2.5.3. O seu código fonte pode ser consultado através do endereço: *https://gitlab.com/inacio.bruno/metadataanalyser-frontend*. Estas funções têm como objetivo recolher a informação colocada no formulário de envio do ficheiro de metadados e enviá-la à interface computacional, de modo a iniciar o processo de análise e avaliação do mesmo. Para além disso, suportam também o modelo de *Long Polling*, implementado pela interface computacional, de modo a informar o utilizador do progresso e finalização do processo, que pode ser consultado na Figura 3-4.

Figura 3-22 - Desenho do interface de utilizador da solução da tese.

O desenho da interface, ilustrador na Figura 3-22, foi obtido a partir de um repositório externo (*HTML5Layoutst*[9]) e adaptado às funções que fazem parte da descrição da interface de utilizador descrito na Secção 3.2.1.

3.4 Sumário

Neste capítulo foi descrita, de forma técnica e detalhada, a implementação de uma solução informática para a resolução do problema identificado nesta tese (ver Secção 1.1), de como analisar e avaliar a qualidade de integração semântica de metadados, com base na utilização das medidas de qualidade semântica definidas na Secção 3.1. Estas medidas são utilizadas para a análise de um ficheiro de metadados, através da avaliação dos valores de especificidade e de cobertura das anotações nele encontradas, utilizadas para descrição do conjunto de dados, de modo a aferir o seu grau de integração semântica, ou de qualidade semântica, com o conjunto de ontologias de referência.

[9] http://www.html5layouts.com

Para cumprir com este objetivo foi definida uma arquitetura por camadas de modo a permitir a interação entre o utilizador final, humano ou computacional, e os procedimentos de análise e avaliação necessários ao objetivo. Foram assim definidas quatro camadas: (*i*) a camada de apresentação, (*ii*) a camada *Web*, (*iii*) a camada de análise e (*iv*) a camada de dados.

A camada de apresentação define os interfaces para os utilizadores humanos, a camada *Web* estabelece os interfaces para utilizadores computacionais, a camada de análise apresenta o modelo computacional de análise e avaliação de metadados e a camada de dados engoba os repositórios de ontologias necessárias à avaliação de anotações.

No próximo capítulo é apresentada a avaliação da aplicação deste sistema desenvolvido, através da análise dos resultados de execução do procedimento de cálculo de especificidade, implementado sobre o modelo relacional de cada ontologia, e de um caso de estudo que tem como base um dos repositórios públicos de investigação na área da bioinformática, o *MetaboLights*. Por fim é apresentado o resultado de um questionário de usabilidade destinado a avaliar a interação dos utilizadores com a plataforma.

Capítulo 4

Avaliação das Contribuições

4.1 Procedimento de cálculo de especificidade

De modo a avaliar a precisão do procedimento de cálculo de especificidade de uma anotação, através da indicação do seu URI, foi necessário elaborar um cenário de testes onde fosse possível comparar os valores obtidos pelo procedimento SQL desenvolvido na contribuição da tese, e os valores obtidos num estudo anterior [16]. Neste estudo foi utilizado um algoritmo desenvolvido em *Python* que percorre o repositório *Metabolights* [15], através de sucessivas chamadas HTTP, de modo a obter o detalhe que necessita.

4.1.1 Implementação

Como caso de estudo foram escolhidas cinco ontologias, em formato OWL e TTL, do repositório de ontologias biomédicas *BioPortal* [44], e convertidas para o modelo relacional utilizado na contribuição da tese, descrito na Secção 3.2.4, com auxílio da ferramenta de conversão *OWLtoSQL*, descrita na Secção 2.5.1. Estas foram escolhidas tendo em conta a sua frequente utilização na anotação dos recursos do *MetaboLights*, o número de conceitos e a profundidade que apresentam nas relações entre estes.

Tabela 4-1 - Lista de ontologias para o procedimento de cálculo de especificidade

Ontologia	**Descrição**	**Conceitos**	**Prof.**
BIOMODELS	Ontologia de modelos no repositório *BioModels*	187519	27
ICD10CM	Sistema para classificação e codificação de diagnósticos, sintomas e procedimentos clínicos	92168	6
SNMI	Ontologia de nomenclaturas sistematizadas de medicina humana e veterinária	109150	7
OMIM	Ontologia sobre Herança Mendeliana no Homem	81820	2
VTO	Ontologia sobre taxonomia de vertebrados	107134	38

Por cada ontologia, descritas na Tabela 4-1, foram escolhidos dez conceitos tendo como critério o seu posicionamento num determinado ramo da representação em árvore invertida da ontologia. A lista de todos os conceitos pode ser consultada no Anexo H. Foram escolhidos os conceitos que não se situam quer no topo, quer no fundo do ramo a que pertencem, i.e., foram escolhidos conceitos que apresentam ascendentes e

descendentes. Por cada uma foi (*i*) feita uma chamada ao procedimento SQL *sp_conceptspec*, descrito na Secção 3.2.4, e (*ii*) executado o procedimento em *Python* utilizado no estudo [16]. Para cada um destes ambientes foi registado o número de conceitos ascendentes e descendentes folha encontrados no conceito analisado, assim como o tempo de execução de cada chamada e respetivo valor de especificidade obtido. A unidade de tempo encontra-se em milissegundos e o intervalo de especificidade, S, entre $0 \leq S \leq 1$. O valor 0 significa o menor valor semântico possível pois o conceito encontra-se no topo da árvore, a melhor descrição possível. O valor 1 significa o maior valor semântico possível pois não existem mais conceitos ao longo do ramo da árvore que possam oferecer uma descrição melhor.

4.1.2 Resultados obtidos

Os resultados obtidos podem ser consultados no Anexo H. Estes revelam que (*i*) o número de ascendentes e número de descendentes folha de cada um dos conceitos em estudo é exatamente igual para ambos os procedimentos, (*ii*) o valor de especificidade apresenta-se igual em todas as anotações com diferenças inferiores a 0,0001 (não é exatamente 0 por questões de arredondamento), (*iii*) o tempo médio de execução do procedimento SQL representa apenas 0,0037% do tempo médio de execução do procedimento *Python* desenvolvido no estudo [16]. Para cada um dos fatores foi calculada a média aritmética.

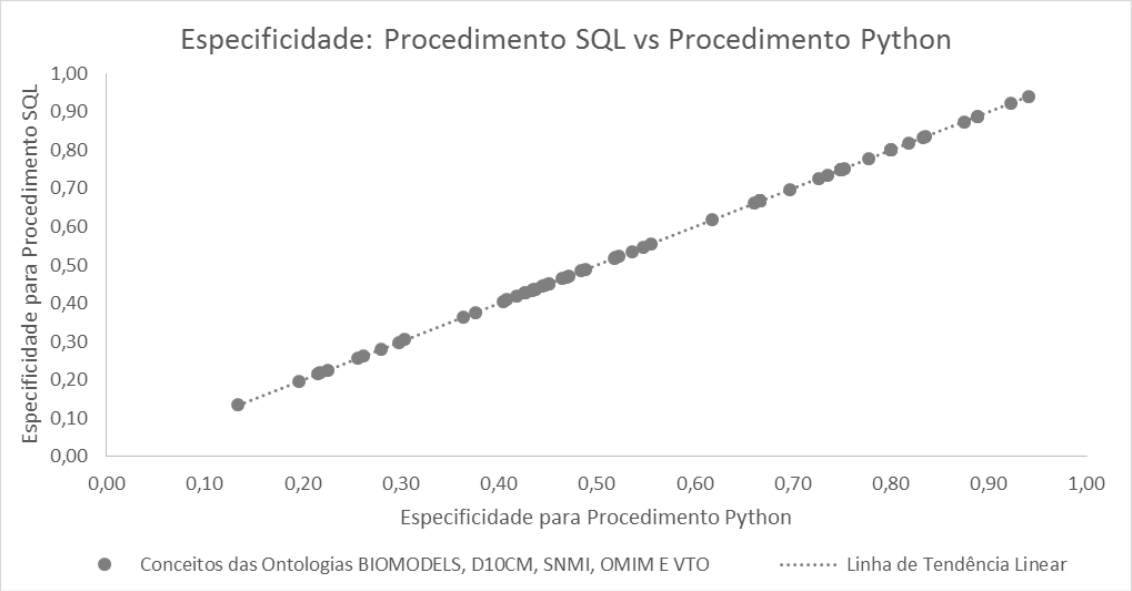


Figura 4-1 - Correlação de especificidade entre SQL e Python.

Este procedimento *Python* efetua o cálculo de especificidade de anotações através de chamadas HTTP a um interface computacional, disponibilizado pelo repositório *MetaboLights*. Este interface é acedido através de um ponto único (http://data.bioontology.org/ontologies), que disponibiliza um dicionário onde cada

entrada corresponde a uma ontologia. A navegação pelos conceitos das ontologias é feita através de URI, sendo que cada chamada retorna uma vista estruturada (em formato JSON) das propriedades e relações com outros conceitos (ascendentes e descendentes), do conceito em consulta. Este é naturalmente um processo demorado, pois por cada passo de navegação é necessária uma nova chamada HTTP. É assim normal que o tempo de cálculo de especificidade de um conceito seja demorado.

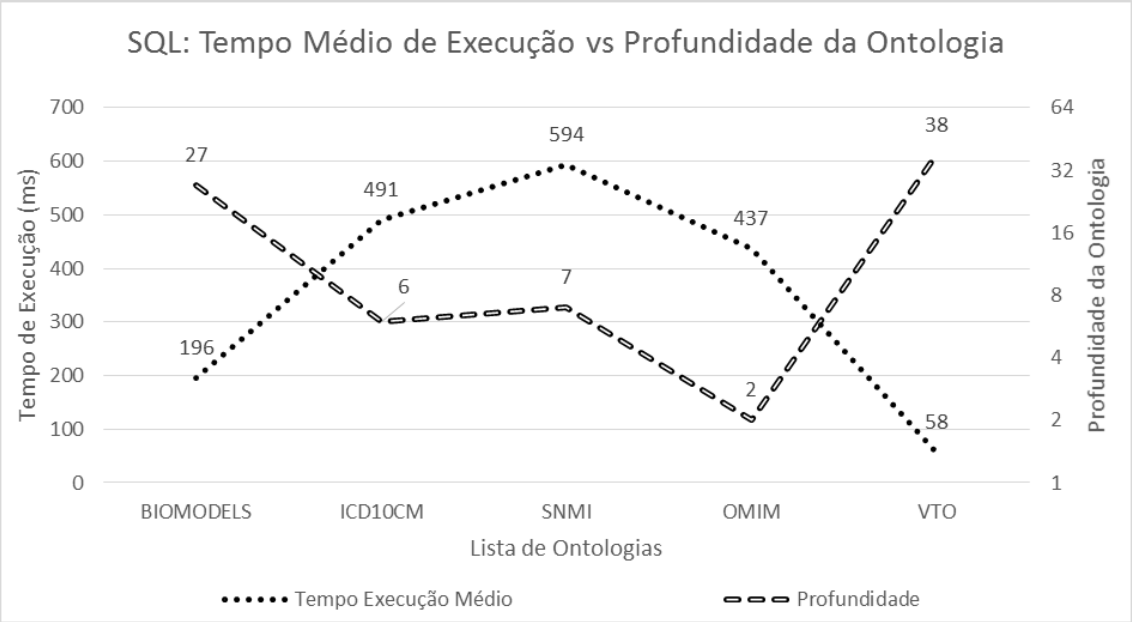


Figura 4-2 - Correlação entre tempo de execução e profundidade das Ontologias.

Tendo em conta todos os conceitos utilizados para cálculo de especificidade, foram encontradas algumas correlações que importa realçar. Foi encontrada uma forte correlação positiva com um valor de 0,99 na relação entre os valores de especificidade de cada conceito dos procedimentos *Python* e SQL, que pode ser consultado na Figura 4-1.

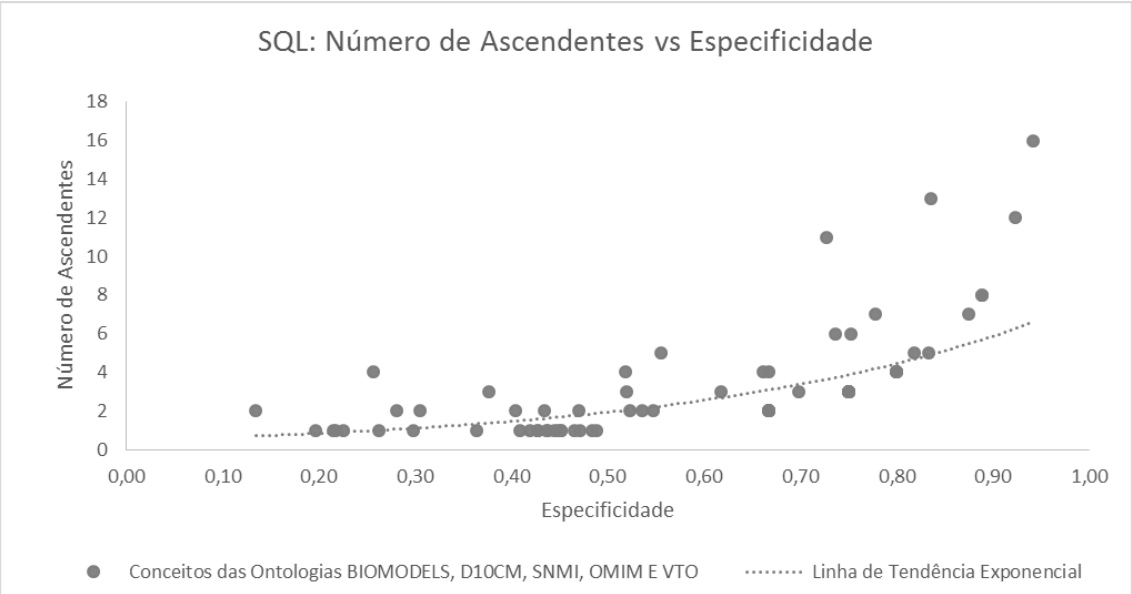


Figura 4-3 - Correlação entre número de ascendentes e especificidade.

Foram também encontradas algumas correlações, tendo em conta o número de conceitos e os níveis de profundidade existentes em cada ontologia, descritas na Tabela 4-1, e as médias de tempo de execução do procedimento SQL. Uma das correlações diz respeito à profundidade das ontologias, que apresenta uma forte correlação negativa, com o valor de -0,93, com o tempo médio de execução do procedimento SQL no cálculo de especificidade, ilustrada na Figura 4-2.

Outra correlação do procedimento SQL diz respeito ao número de ascendentes de cada conceito, que apresenta uma correlação positiva de 0,65 com os valores de especificidade obtidos, ilustrada na Figura 4-3, o que indica que à medida que o número de ascendentes de um conceito aumenta, mais profunda é a sua posição num determinado ramo da árvore, o que corresponde a uma maior especificidade.

A última correlação encontrada diz respeito ao valor de descendentes folha, que possui uma correlação negativa com o valor de especificidade calculado para cada conceito, de -0,51. Esta correlação revela que à medida que o número de descendentes folha aumenta, i.e., descendentes que se situam no fundo da árvore da ontologia, a especificidade do conceito diminui (correlação ilustrada na Figura 4-4). Isto indica que quanto maior for o número de descendentes mais alta é a posição da conceito no ramo e logo menor é a sua especificidade semântica. Esta correlação encontra-se em linha com a anterior. Quanto maior for o numero de ascendentes, maior será a especificidade. Quanto maior for o número de descendentes folha, menor será a especificidade.

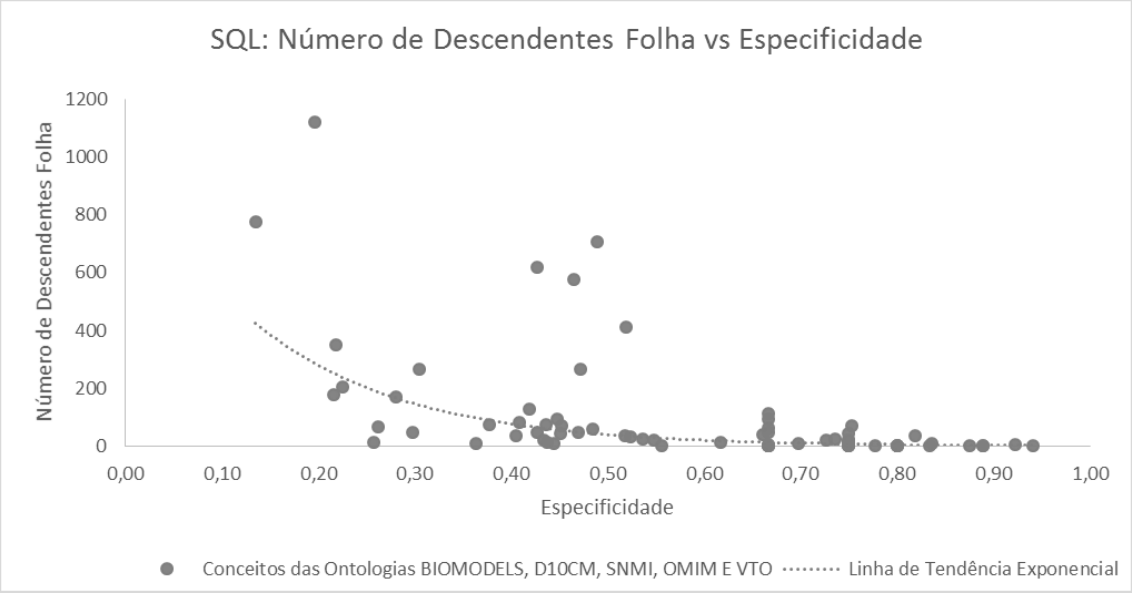


Figura 4-4 - Correlação entre número de descendentes folha e especificidade.

4.1.3 Discussão

De acordo com os resultados obtidos é possível concluir que o procedimento SQL desenvolvido, de modo a implementar as medidas de avaliação de qualidade de integração semântica, descrita na Secção 3.1, apresenta resultados idênticos ao algoritmo implementando no estudo [16], com uma diferença de valores de especificidade inferior a 0,0001. A partir das correlações entre vários fatores (especificidade, tempo de execução, número de ascendentes e número de descendentes folha) é possível concluir que os resultados do procedimento SQL apresentam uma base correta de cálculo da especificidade, tendo em conta a metodologia definida no estudo de medidas de qualidade de integração semântica, que supõe este valor como a relação entre o número de ascendentes e descendentes folha de um dada conceito, em um dado ramo da árvore da ontologia a que pertence.

Um maior número de ascendentes prossupõe uma maior especificidade, i.e., um ganho na atribuição semântica da anotação de metadados, pois a posição do conceito no ramo da árvore da ontologia encontra-se potencialmente mais abaixo. Pelo contrário um maior número de descendentes prossupõe uma menor especificidade, e um menor ganho na atribuição semântica da anotação, pois a posição do conceito no ramo está potencialmente mais acima e logo mais distante do fundo da árvore.

4.2 Caso de Estudo Metabolights

De modo a ser possível uma avaliação da solução, que foi desenvolvida como elemento da contribuição desta tese, foi necessária a especificação de um caso de utilização que implicou a adequação da infraestrutura a um repositório de dados científicos particular, cujos metadados utilizados para a caraterização dos dados armazenados estivessem já integrados semanticamente com ontologias externas. Para o efeito foi considerado o repositório *Metabolights* [15] que apresenta uma base de dados para experiências na área de *Metabolomics* e dados derivados. A sua implementação baseia-se na especificação ISA-TAB, detalhada na Secção 2.4.1

A área de *Metabolomics*, é um estudo de larga escala de pequenas moléculas, também conhecidas como *metabolites* (metabólitos), dentro de células, biofluídos, tecidos e organismos [78]. Esta área é dividida em quatro subáreas: (*i*) *genomics*, (*ii*) *transcriptomics*, (*iii*) *proteomics* e (*iv*) *metabolomics*.

Como procedimento de avaliação, foram analisados e classificados todos os ficheiros de metadados existentes no repositório (161) através do motor de análise e avaliação. Posteriormente foram selecionados os metadados considerados mais relevantes do ponto de vista da avaliação global de especificidade das anotações e cobertura de termos. Foram escolhidos metadados tendo em conta a variação da relação entre especificidade e

cobertura, no espetro dos resultados (baixa, intermédia e alta). Para cada um foi elaborada uma verificação manual dos valores apresentados. Por fim, os resultados foram comparados, pelo autor desta tese, com aqueles obtidos pelo estudo anterior [16]. Foram também posteriormente adicionados, metadados de mais três estudos do repositório que não estavam, no momento da análise global, disponíveis para consulta: (*i*) o MTBLS286, (*ii*) o MTBLS287 e (*iii*) o MTBLS288. A particularidade destes, em relação aos demais, centra-se na disponibilidade da equipa, que mantém o repositório *Metabolights*, em fornecer as versões originais (submetidas pelos investigadores) de cada um dos ficheiros de metadados, proporcionado assim um ambiente de comparação pré e pós revisão manual, feita no próprio repositório.

De modo geral, este caso de estudo baseia-se nas duas medidas, descritas na Seção 3.1: (*i*) especificidade e (*ii*) cobertura. A especificidade mede o nível de conhecimento proporcionado por uma anotação, com referência a um conceito ontológico, para descrição de uma propriedade dos metadados. A cobertura mede a razão entre anotações com referência a conceitos ontológicos e o número total de anotações utilizadas nos metadados.

4.2.1 Recolha de resultados

Uma vez configurado o motor de análise e avaliação, de acordo com a especificidade do repositório *Metabolights* [15] e convertidas todas as ontologias necessárias, foi desenvolvido um projeto Java paralelo no sentido de desenvolver um cliente direto do motor. Durante a sua execução foram enviadas mensagens sequenciais, onde em cada uma constava um ficheiro de metadados diferente. O resultado da análise e avaliação, como descrito na Secção 3.2.3, de cada um dos metadados foi salvaguardado num ficheiro com o nome do estudo respetivo. Estes resultados foram posteriormente compilados num único ficheiro, para melhor compreensão.

No momento de elaboração deste estudo, o repositório *Metabolights* é composto por cento e sessenta e quatro estudos, salvaguardados através do padrão ISA-TAB. Por cada estudo foi considerado o ficheiro i_Investigation.txt [42] que contém os metadados de cada estudo.

4.2.2 Resultados obtidos

A totalidade dos resultados obtidos pode ser consultada no Anexo J (lista da especificidade por estudo), no Anexo K (lista da cobertura de termos por estudo), no Anexo L (lista de tempos de execução por estudo, em milissegundos) e no Anexo M (lista de todas as anotações encontradas). Do total de ficheiros de metadados analisados (161), apenas 6 (3,72%) não apresentam quaisquer anotações, i.e., o valor de média de

especificidade é 0. Não seria significativo para a média global de especificidade do repositório, que é de 0,81, se o valor mais baixo de especificidade, maior do que zero, não fosse de 0,56 (o estudo MTBLS147). Acaba, no entanto, por ter algum impacto negativo na média de especificidade global do repositório, que sem considerar os estudos com especificidade nula seria de 0,8436, uma diferença de 0,03.

O valor mais alto de especificidade é de 1, que indica a melhor especificidade possível, mas apenas 7 (4,34%) dos metadados apresentam esta especificidade máxima. A média de especificidade do repositório é relativamente alta, já referida de 0,8121, o que revela algum cuidado por parte dos investigadores na escolha semântica das anotações que acompanham os termos.

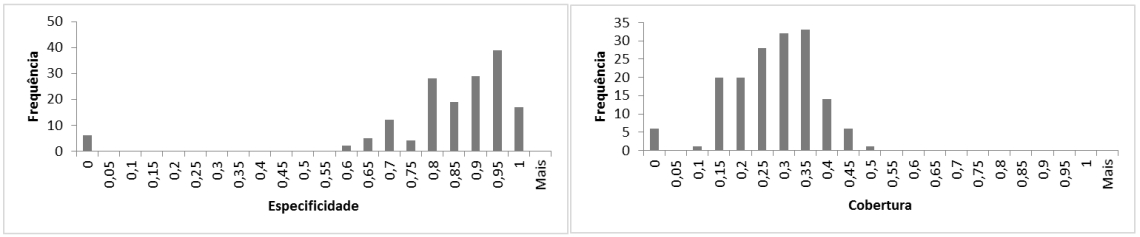


Figura 4-5 - Histograma das médias de especificidade e cobertura de Metabolights.

A média de cobertura de anotações, por termos semânticos, do repositório é de 0,25. De notar que 11 (6,21%) dos 161 metadados apresentam uma média de cobertura nula, e que destes, 6 apresentam também uma especificidade nula, o que indica que à exceção destes foi possível calcular a especificidade de pelo menos uma anotação por cada estudo. O valor mais baixo de cobertura, maior do que zero, é de 0,09 (o estudo MTBLS124). O maior é de 0,47 (o estudo MTBLS148). Apenas 9 (5,59%) dos metadados apresentam uma cobertura de anotações superior a 0,4.

De realçar que apenas foram contabilizadas, para cálculo da cobertura, as anotações para as quais foi possível obter um valor de especificidade, i.e., que façam referência a um conceito ontológico. Dos 368 conceitos ontológicos, encontrados nos metadados do repositório (disponíveis para consulta no Anexo M), 270 (73,36%) não têm um valor de especificidade calculado. No entanto, estes 270 conceitos ontológicos apenas foram utilizados em 29,88% das anotações. De modo geral, o intervalo de valores revela que houve um cuidado de modo geral em descrever as propriedades dos metadados, mas uma igual dose de esforço não foi aplicada na anotação dos mesmos através de ligações a ontologias externas.

As distribuições de frequência das médias de especificidade e de cobertura dos metadados do repositório *Metabolights* são assimétricas, como pode ser verificado na Figura 4-5. A distribuição da média da especialidade apresenta um enviesamento à esquerda, uma vez que a média é menor que a mediana, respetivamente 0,81 e 0,85. Como

50% das anotações estão acima de 0,85 podemos afirmar que estas são de facto informativas, do ponto de vista semântico. A distribuição da média de cobertura apresenta também um ligeiro enviesamento à esquerda. A média é de 0,25 e a mediana é de 0,27. No entanto, 50% dos metadados tem uma cobertura inferior a 0,27, o que representa de modo geral uma muito fraca cobertura de anotações por parte de ligações a conceitos de ontologias.

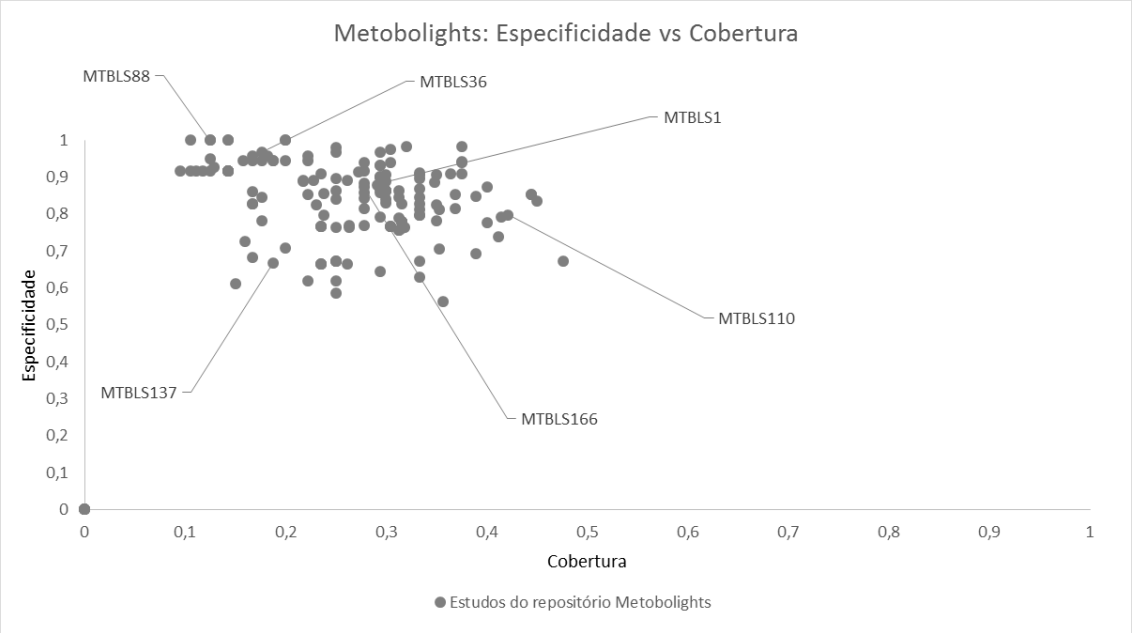


Figura 4-6 - Correlação das médias de especificidade e cobertura de Metabolights.

Foi encontrada uma pequena correlação positiva entre as médias de especificidade e cobertura de 0,32 (com um p-value = 0.000033, considerado estatisticamente significado), o que poderá denotar que os investigadores que mais esforço dedicam à procura da especificidade de termos semânticos, revelam também alguma preocupação e aplicação de esforço na procura de referência a conceitos ontológicos, para um maior número de anotações dos metadados. Tratando-se o *Metabolights* de um repositório real, de apoio à área de *Metabolomics*, é possível concluir que a qualidade de integração semântica dos metadados existentes é fraca, tendo em conta as medidas de avaliação estudadas na Secção 3.1, corroborando por isso o problema apresentado nesta tese.

Conjunto de estudos mais relevantes

Do ponto de vista dos objetivos deste caso de estudo, foi necessário encontrar um conjunto de estudos mais relevantes, do repositório *Metabolights*, a partir dos quais fosse possível comparar valores obtidos com uma revisão manual e também com os valores obtidos pelo procedimento de cálculo implementado no estudo [16]. Como fator de relevância foi considerada a correlação entre as médias de especificidade e cobertura de cada estudo, representada no gráfico de dispersão ilustrado na Figura 4-6.

Tabela 4-2 - Lista de especificiddade e cobertura para o subconjunto de estudos.

	Motor Análise Avalia.		**Procedimento *Python***		**Procedimento Manual**	
Estudo	**Especif.**	**Cobert.**	**Especif.**	**Cobert.**	**Especif.**	**Cobert.**
MTBLS1	0,88	0,3	0	0	0,89	0,3
MTBLS36	0,96	0,17	0	0	0,96	0,17
MTBLS88	0,75	0,31	0,69	0,75	0,75	0,31
MTBLS110	0,91	0,14	0,87	0,5	0,84	0,28
MTBLS137	0,94	0,2	0,87	0,37	0,94	0,2
MTBLS166	1	0,14	0	0,54	0,6	0,23
Média:	0,91	0,21	0,40	0,36	0,83	0,25

O gráfico de dispersão da Figura 4-6 confirma a preocupação em colocar anotações informativos nos metadados por parte dos investigadores, de modo geral, mas em muito pouco número. Este apresenta um aglomerado na fronteira acima dos 50% de especificidade e abaixo dos 50% de cobertura, cujo centro se situa no ponto cartesiano (0,27, 0,85). A amostra de estudos escolhida, listada na Tabela 4-2, teve em conta os critérios já descritos na introdução da avaliação, e demonstrada na Figura 4-6. Foram principalmente considerados estudos posicionados em cada um dos quadrantes do aglomerado.

Tabela 4-3 - Lista de anotações encontradas e avaliadas do subconjunto de estudos.

	Motor Análise Avalia.		**Procedimento Python**		**Procedimento Manual**	
Estudo	**Termos Semânticos**	**Anota.**	**Termos Semânticos**	**Anota.**	**Termos Semânticos**	**Anota.**
MTBLS1	6	20	0	0	6	20
MTBLS36	3	17	0	0	3	17
MTBLS88	5	16	7	16	5	16
MTBLS110	2	14	4	14	4	14
MTBLS137	3	15	3	15	3	15
MTBLS166	3	21	5	21	5	21
Soma:	22	103	19	66	26	103

Os resultados obtidos perante esta amostra de estudos podem ser consultados no Anexo I. O motor de análise e avaliação apresenta um total de anotações, encontradas nos metadados, idêntica ao procedimento manual (apresentada na Tabela 4-3), de 103, muito acima dos 66 do procedimento *Python* utilizado no estudo [16]. O que revela, necessariamente, que o procedimento de análise do motor capturou corretamente os termos existentes nesta amostra. O motor de análise e avaliação apresenta assim uma performance de análise muito semelhante ao procedimento manual, na pesquisa de anotações utilizadas nos metadados, claramente acima do procedimento *Python*.

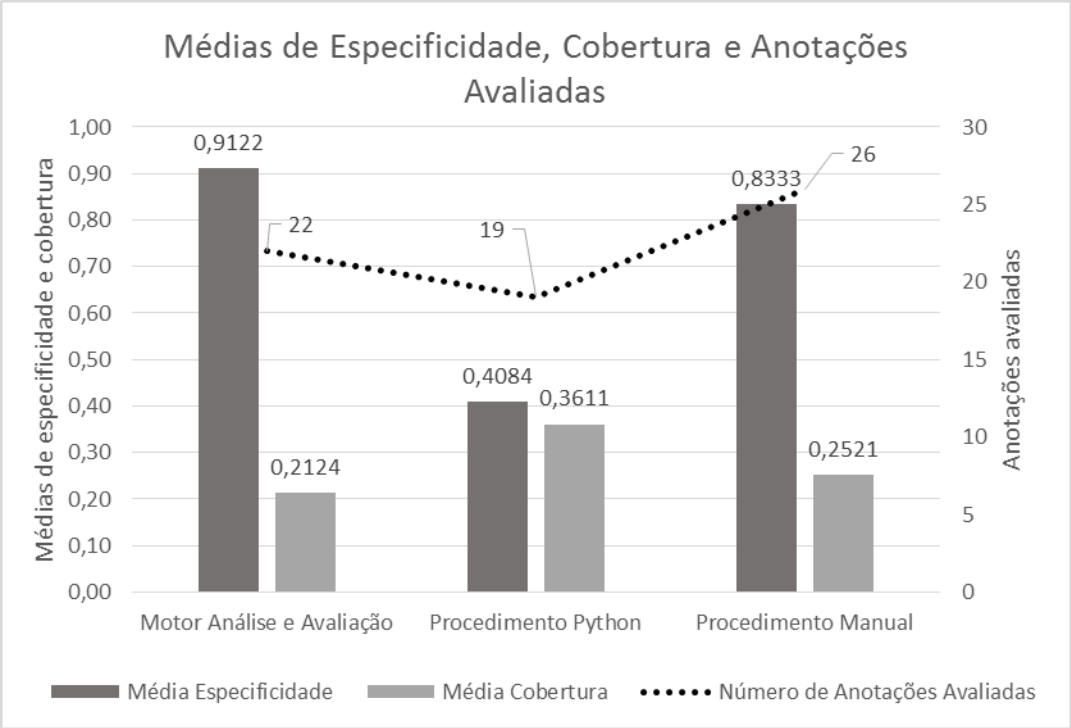


Figura 4-7 - Médias de especificidade, cobertura e anotações avaliadas.

Como pode ser consultado no diagrama da Figura 4-7 (que relaciona os valores apresentadas na Tabela 4-2 e Tabela 4-3), o motor de análise e avaliação apresenta um total de anotações avaliadas, i.e., anotações para as quais foi possível calcular um valor de especificidade, de 22. Este valor representa apenas 21% do total de anotações existentes, cerca de 103. No entanto, através do procedimento manual também apenas foi possível avaliar 26 (25%) das 103 anotações encontradas. O procedimento *Python* avaliou 19 (28%) das anotações 66 existentes.

Estes valores corroboram os resultados obtidos anteriormente sobre o repositório *Metabolights*, apresentando sobretudo uma fraca cobertura dos metadados utilizados neste repositório, desta feita validados através de uma verificação manual. O motor de análise e avaliação conseguiu avaliar 22 anotações, ou termos semânticos, das 26 avaliadas pelo procedimento manual. Este facto ficou a dever-se, sobretudo, à

incompletude do repositório de ontologias, que não engloba todas aquelas que são necessárias no âmbito destes estudos. Ainda assim, obteve um resultado acima do procedimento *Python*.

Com base nestas anotações, o motor de análise e avaliação obteve um melhor resultado de média de especificidade, de 0,91 para 22 anotações avaliadas, acima do procedimento manual com 0,83, para 26 anotações avaliadas, como pode ser verificado na Tabela 4-2 e na Figura 4-7. No entanto, este resultado foi obtido porque não foram avaliadas algumas anotações de fraca especificidade, dos estudos MTBLS110 (2 anotações) e MTBLS166 (2 anotações), que potencialmente fariam baixar a média de especificidade obtida pelo motor de análise e avaliação. A ontologia às quais as anotações se referiam não estava presente no repositório do motor de análise e avaliação. O procedimento *Python*, por sua vez, apresenta uma fraca média de especificidade, de 0,40, para 19 anotações avaliadas, sobretudo porque não encontrou uma parte considerável das anotações existentes nos metadados, e para aquelas que encontrou atribui-lhes um valor tendencialmente mais baixo de especificidade. Isto deveu-se sobretudo à implementação do algoritmo de navegação do repositório de metadados. A interface de consulta disponibilizada pelo repositório agrupa os resultados em grupos de 50 elementos, aos quais dá o nome de páginas. Na pesquisa por termos descendentes, daquele que se encontra em avaliação, o procedimento *Python* apenas considera a primeira página de resultados para cálculo da especificidade do termos dado (um menor número de descendentes equivale a um maior valor de especificidade). Se o termo tiver um número superior de descendentes, na ontologia em avaliação, não são simplesmente considerados pelo procedimento. Por sua vez, o motor de análise e avaliação tem em consideração todos os descendentes do termo em avaliação, possibilitando assim um cálculo mais preciso da sua especificidade.

Do ponto de vista das médias de cobertura de anotações, estas apresentam valores baixos em todos os procedimentos (motor, *Python* e manual). Os valores de cobertura do procedimento manual estão de acordo com o padrão de dispersão da Figura 4-6, que apesar de referir apenas valores de especificidade e cobertura obtidos pelo do motor de análise e avaliação, permite reafirmar que a preocupação da descrição semântica, do ponto de vista quantitativo, dos metadados, por parte dos investigadores, aparenta ser baixa.

Os valores médios de cobertura de anotações são muito próximos, entre o motor de análise e o procedimento manual, com uma média de cobertura de 0,21 e 0,25, respetivamente, de acordo com a Tabela 4-2. Apenas os estudos MTBLS110 e MTBLS166 diferem nos valores de cobertura, porque o motor não foi capaz de avaliar 4 das anotações existentes. Por seu lado, o procedimento *Python* apresenta uma média de

cobertura superior a todos os procedimentos, de 0,36 (de acordo com a Tabela 4-2), isto apesar de não ter sido capaz de encontrar quaisquer anotações para os estudos MTBLS1 e MTBLS36. Para os restantes estudos deste procedimento, os valores de cobertura são bastante superiores aos dos restantes procedimentos.

Estes valores mais altos devem-se, sobretudo, ao método utilizado para cálculo da cobertura do procedimento *Python*. Neste, para cálculo da cobertura dos metadados é considerada a média de cobertura de cada uma das classes de anotação, existentes nos metadados, cujo valor de cobertura seja superior a 0, i.e., se para uma, ou mais classes, apesar de englobarem anotações não foi possível calcular a especificidade de nenhuma delas, esta classe não é considerada para a cobertura global.

Do ponto de vista destes resultados, é possível afirmar: (*i*) que o desempenho, na avaliação da qualidade de integração semântica de metadados com base nas medidas definidas na Secção 3.1, do motor de análise e avaliação encontra-se muito próximo de uma análise e avaliação manual e com uma prestação superior a procedimentos de avaliação da qualidade de integração semântica de metadados estudados anteriormente, tendo em conta o espetro da relação entre especificidade e cobertura do repositório *Metabolights*; (*ii*) que através de uma maior completude por parte do repositório de ontologias, utilizada pelo motor para cálculo da especificidade das anotações encontradas, é possível obter resultados tendencialmente mais próximos daqueles obtidos por uma análise e avaliação manual.

Estudos MTBLS286, MTBLS287 e MTBLS288

Para os estudos MTBLS286, MTBLS287 e MTBLS288 (adicionados posteriormente ao caso de estudo) foi possível analisar e avaliar os metadados originalmente submetidos pelos investigadores, com já referido na introdução da avaliação. A descrição inicialmente feita por estes revela-se, no entanto, fraca, como pode ser consultada na Tabela 4-4 (secção Anterior à Revisão Manual). Para além do baixo número de anotações, a soma do número de anotações com referência a conceitos ontológicos (4) representa apenas 14% da soma do número de anotações totais (27), utilizadas para descrição dos estudos.

Tabela 4-4 - Anotações e termos dos estudos MTBLS286, MTBLS287, MTBLS288.

	Anterior à Revisão Manual		**Posterior à Revisão Manual**	
Estudos	**Termos Semânticos**	**Anotações**	**Termos Semânticos**	**Anotações**
MTBLS286	0	9	5	16
MTBLS287	2	9	5	16

MTBLS288	2	9	5	16
Soma:	4	27	15	48

No entanto, o processo de revisão manual (cura) permitiu um aumento de 200% (de 27 para 48) no número de anotações utilizadas para descrição dos metadados, assim como um aumento de 375% (de 4 para 15) de anotações com referência a conceitos ontológicos, como pode ser consultado na Tabela 4-4 (secção Posterior à Revisão Manual), o que se traduz em um aumento de 17 pontos percentuais na média de anotações utilizadas para cobertura de termos, de 14% para 31%.

Os valores de especificidade e cobertura de cada um dos estudos, ilustrados na Figura 4-8, demonstram que um dos maiores ganhos na ação de revisão manual, para além da utilização de novas anotações para descrição dos metadados, é o aumento da cobertura de anotações, de modo geral, por anotações com referência a conceitos ontológicos, um procedimento essencial à integração semântica de metadados e aumento da sua qualidade. A cobertura média destes estudos, anterior à revisão, é de apenas 0,14. O valor após revisão manual é de 0,25, muito perto da média de cobertura do próprio repositório, também ela de 0,25.

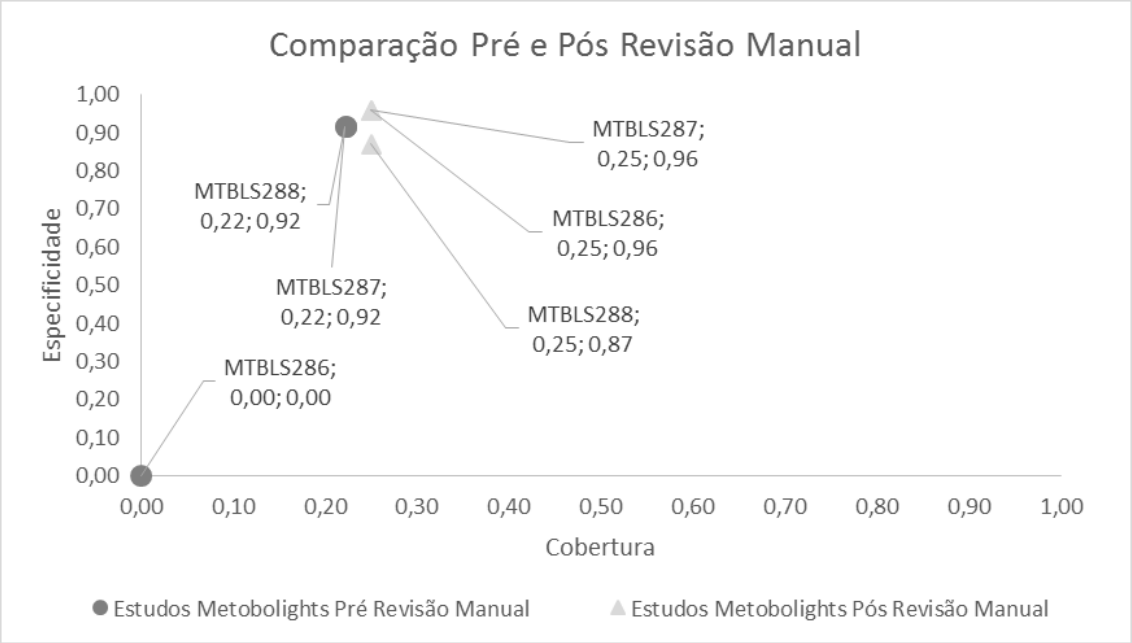


Figura 4-8 - Resultados para os estudos MTBLS286, MTBLS287 e MTBLS288.

Para esta diferença entre períodos muito contribui o estudo MTBLS286, onde inicialmente não foram utilizadas quaisquer anotações (evolução ilustrada na Figura 4-8), mas que após a revisão passou para uma especificidade de 0,96 e cobertura de 0,25. Curiosamente, os valores de especificidade para os estudos MTBLS287 e MTBLS288 apresentam-se acima da média do repositório *Metabolights*, de 0,81, em ambos os períodos (pré e pós revisão manual), com uma diferença inferior a 0,05. Isto revela que a

atenção dada à escolha de anotações com referência a termos de ontologia foi alta, por parte dos investigadores, apesar do seu baixo número, um facto comum a todo o repositório. Os valores médios de especificidade e cobertura, destes três estudos, enquadram-se no padrão de valores obtidos para os restantes estudos do repositório *Metabolights*.

De modo geral, o processo de cura feito pela equipa do repositório traduziu-se essencialmente em uma melhoria do ponto de vista quantitativo de anotações, da descrição dos metadados destes três estudos. À exceção do estudo MTBLS286, que sofreu de facto uma melhoria, quer da sua especificidade, quer da sua cobertura de anotações, os restantes viram aumentado o número de anotações utilizadas para descrição dos seus metadados, e proporcionalmente um aumento da utilização de termos semânticos, utilizados para referência a conceitos ontológicos, i.e., para os estudos MTBLS287 e MTBLS288 não existiu propriamente uma melhoria assinável na qualidade de integração semântica dos seus metadados visto que, quer o valor de especificidade, quer o valor de cobertura apenas sofreram pequenas variações: (*i*) de 3 e 4 pontos percentuais, respetivamente, para o estudo MTBLS287; (*ii*) de 3 e -5 pontos percentuais para o estudo MTBLS288. Apesar de pequenas, a plataforma desenvolvida nesta tese foi capaz de identificar todas as melhorias implementadas no processo de cura, demonstrando assim a precisão do motor de análise e avaliação no processamento destes três ficheiros de metadados.

4.2.3 Discussão

A avaliação da integração semântica de metadados apresenta um conjunto de desafios que são tecnologicamente difíceis de ultrapassar. Em primeiro lugar é necessário ler e extrair os metadados que acompanham os conjuntos de dados, de modo a extrair os elementos utilizados para a integração semântica feita pelos investigadores. Cada repositório tem um formato específico de salvaguarda dos metadados, o que dificulta à partida a criação de um meio geral de avaliação.

O desenho da arquitetura do motor de análise e avaliação, um dos elementos essenciais na contribuição desta tese, coloca grande ênfase na adaptabilidade da solução, de acordo com os requisitos não-funcionais da Tabela 3-3, de forma a permitir que cada repositório possa ser interpretado de forma individual, de modo a que anotações, com e sem referência a conceitos ontológicos, possam ser corretamente encontradas e extraídas dos metadados que acompanham os dados presentes em cada um. Em segundo lugar é necessário avaliar cada uma das anotações, do ponto de vista da sua integração semântica, com recurso à ontologia à qual pertence, através da implementação do estudo de medidas de qualidade de integração semântica feito como contribuição desta tese, detalhado na Secção 3.1. Para que isso seja possível é por sua vez necessário a inclusão das ontologias

referidas no repositório da solução. Quanto mais completa for esta inclusão melhores serão os resultados do motor de análise e avaliação.

Do ponto de vista dos resultados obtidos, e tendo em conta os desafios colocados na avaliação da integração semântica e também na sua implementação por parte dos investigadores responsáveis pelos dados e correspondentes metadados, é possível chegar a duas conclusões: (*i*) o desempenho, na avaliação da qualidade de integração semântica de metadados, do motor de análise e avaliação pode ser considerado positivo, especialmente em comparação com a implementação anterior feita em *Python* e com recurso a uma API através de HTTP, pois conseguiu apresentar valores de especificidade e cobertura tendencialmente próximos de uma verificação manual, e sempre superiores a outros procedimentos de avaliação anteriores; (*ii*) os resultados de avaliação do repositório *Metabolights* corroboram a existência do problema identificado na introdução desta tese, de forma quantificável e objetiva, podendo servir como ponto de partida para futuros trabalhos que pretendam motivar os donos da informação a anotar os seus dados.

Desempenho do motor de análise e avaliação

O desempenho do motor de análise e avaliação pode ser medido tendo em conta a performance do procedimento de análise em relação ao número de anotações encontradas, descrito na Secção 3.2.3.3, e a performance do procedimento de avaliação tendo em conta o número de anotações para as quais foi possível calcular um valor de especificidade e nível de precisão respetivo, descrito na Secção 3.2.3.6. Os resultados obtidos na análise da amostra de estudos do repositório *Metabolights* [15] (MTBLS1, MTBLS36, MTBLS88, MTBLS110, MTBLS137 e MTBLS166), apresentam valores idênticos, na identificação de termos (num total de 103) e anotações (num total de 30), à analise manual dos estudos e em alguns casos superiores ao procedimento adaptado pelo estudo anterior (num total de 19 anotações e 66 termos) do repositório [16], o que permite afirmar que o procedimento de análise implementado no motor de análise e avaliação identifica claramente as anotações utilizadas nos metadados. Este é um procedimento essencial no processo de avaliação da qualidade de integração semântica de metadados, pois sem uma correta identificação de anotações não seria possível obter valores de cobertura de anotações por parte daquelas com referência a conceitos ontológicos, nem de especificidade destas.

O procedimento de avaliação do motor de análise e avaliação apresenta valores diferentes, quer do procedimento manual, quer do procedimento adotado pelo estudo [16], com médias de especificidade respetivamente de 0,91, 0,83 e 0,40, na amostra de estudos referida anteriormente. Como foi já referido, a avaliação das anotações depende da cobertura de ontologias presentes no repositório de ontologias da plataforma. No entanto, o padrão de valores de avaliação individual das anotações, implementado através do

algoritmo de avaliação de metadados detalhado na Secção 3.1, está de acordo com os valores apresentados, quer pelo procedimento manual, quer pelo procedimento *Python* do estudo já referido. Caso seja possível colocar todas as ontologias referenciadas pelas anotações no repositório, o resultado da especificidade será tendencialmente próximo da avaliação manual. No entanto, nem mesmo através do procedimento manual foi possível avaliar todas as anotações, pois algumas delas referem-se a ontologias que não estão publicamente acessíveis.

O sucesso da avaliação da especificidade dos metadados está diretamente ligado ao cuidado com o qual os investigadores escolhem os termos utilizados na descrição dos mesmos, i.e., o cuidado aplicado na integração dos seus dados com conceitos presentes em domínios de conhecimento públicos e aceites pela sua comunidade, um dos problemas enunciados nesta tese. O desempenho do motor de análise e avaliação encontra-se não apenas em linha com aquele que foi obtido pelo procedimento *Python* implementado no estudo [16], como o suplanta do ponto de vista da qualidade das avaliações feitas sobre as anotações encontradas.

Fraca integração semântica dos metadados

Os resultados obtidos na avaliação do motor de análise e avaliação permitem também validar a hipótese apresentada no problema descrito nesta tese. A possível causa da fraca tendência de integração e partilha de dados é sobretudo social, muito mais do que tecnológica. A relação entre cobertura e especificidade, representada através do gráfico de dispersão na Figura 4-5, apresenta claramente uma tendência de descrição semântica de dados, no repositório *Metabolights* [15], para ser específica mas pouco abrangente, i.e., existe claramente um esforço em descrever as várias classes de anotação dos metadados que identificam os dados, através de um conjunto considerável de conceitos, mas apenas 29% são anotados por referências a termos de ontológicos, com uma especificidade superior a 0,81 em 64% dos casos.

Tendo em conta que para este repositório, em particular, existem ferramentas especializadas no auxílio aos investigadores, descritas na Secção 2.4, à criação, à integração e partilha de metadados com um forte enquadramento semântico, é possível afirmar que a fraca abrangência da descrição semântica empregue pelos investigadores na criação dos seus metadados se deve sobretudo a questões de índole social, que poderão ter como base [3] a falta de destreza em trabalhar com as ferramentas à disposição, o desconhecimento do domínio de conhecimento a conceitos utilizar, a consideração que a partilha não é relevante para o processo cientifico, ou que envolve demasiado trabalho para o benefício que traz.

O que ficou realçado neste caso de estudo foi que a solução pensada para a contribuição que esta tese fez sobre o problema enunciado produz valores muito próximos de uma avaliação manual, e superiores em termos de abrangência a estudos anteriores. Evidencia também que a integração semântica dos metadados se encontra num estado ainda pouco esclarecido, revelando-se por isso como uma ferramenta essencial na medição da qualidade de integração semântica de metadados e com tal um forte complemento ao processo de criação, integração e partilha de metadados no âmbito dos princípios de *Web* Semântica.

4.3 Questionário de usabilidade da solução

Foi elaborado um questionário de usabilidade da solução, que pode ser consultado no Anexo O. Este teve como objetivo avaliar a concretização dos requisitos funcionais e não funcionais da solução, descritos na Tabela 3-2 e Tabela 3-3 respetivamente. Do conjunto de questões colocadas, as mais pertinentes centram-se na navegabilidade da interface desenhado e na facilidade com que o utilizador encontrou a tarefa principal (o envio de um ficheiro de metadados para avaliação), na facilidade de compreensão das metodologias de indicação de metadados e preenchimento do formulário da tarefa principal, na compreensão das etapas do processo de avaliação, no tempo despendido no processo de análise e avaliação dos metadados e sobretudo na compreensão e utilidade dos resultados apresentados pela solução.

O questionário foi enviado ao EMBL-EBI, grupo de investigação responsável pela criação e manutenção do *Metabolights*, de modo a que fosse possível obter algumas respostas de um grupo de pessoas que se encontram envolvidas no processo de integração semântica e partilha de dados através da anotação de metadados, do repositório daquele instituto. Foi notada uma grande aceitação por parte do grupo, que resultou em um conjunto de teleconferências sobre a funcionalidade oferecida pela solução e potenciais formas de integração com outras ferramentas existentes, mas apenas foi possível obter uma resposta ao formulário enviado. No entanto, a resposta dada foi bastante positiva, revelando que os requisitos da solução foram cumpridos.

4.4 Sumário

Neste capítulo foram apresentados os testes desenvolvidos no âmbito da solução desenhada e implementada como contribuição desta tese. Em primeiro lugar foi testada a implementação do algoritmo de medição de especificidade de uma anotação, presente nos metadados. Esta foi comparada com os resultados de um estudo prévio sobre a mesma medida, e os resultados demonstraram que a implementação foi bem-sucedida, pois os valores de especificidade são na sua maior parte idênticos aos anteriores. Em segundo

lugar foi testada a implementação do motor de análise e avaliação através da comparação de resultados obtidos pela análise de todos os metadados existentes num único repositório. Esta comparação foi feita também com recurso aos resultados do estudo anterior e com recurso a uma análise e avaliação manual de uma amostra de estudos do repositório. Em terceiro foi elaborado um questionário de usabilidade da solução, de modo a averiguar a facilidade de utilização da interface de utilizador e da utilidade dos resultados proporcionados

Os resultados permitiram concluir, sobretudo, que o motor efetua uma avaliação de acordo com o estudo das medidas de avaliação de qualidade de integração semântica, apresentado na Secção 3.1, e que a motivação para a integração semântica de metadados está ainda numa fase pouco esclarecida, pelo menos no repositório estudado.

Capítulo 5

Conclusão

Nesta tese são apresentadas ferramentas que auxiliam os investigadores na tarefa de criação, integração e partilha de metadados através da avaliação do nível de conhecimento colocado na descrição dos mesmos. A integração semântica dos metadados fica assim entregue aos esforços que cada investigador considera ser necessários, de modo a proporcionar um maior conhecimento a quem pretende utilizar a sua informação. Através dos resultados obtidos nesta tese foi possível constatar que os esforços colocados na integração e partilha de metadados estão longe de ser os necessários.

A contribuição desta tese pretende ser um ponto de partida para a introdução de um novo conceito que aposta na recompensa e reconhecimento [1] de quem mais esforços dedica à integração e partilha de metadados através dos conceitos de *Web* Semântica, utilizando vocabulários controlados e considerados a norma do seu domínio de conhecimento, pela comunidade científica. A contribuição feita permite que este esforço de integração possa ser medido e avaliado de modo a quantificar a respetiva recompensa e potencial reconhecimento [41].

De modo a assegurar a sua contribuição, esta tese cumpriu com os seguintes objetivos propostos inicialmente:

- **Estudo para avaliação da qualidade da integração na *Web* Semântica**

  Foi desenvolvido um estudo de avaliação da qualidade da integração semântica de metadados através da definição das medidas de especificidade de uma anotação e de cobertura de termos. A medida de especificidade apresenta um valor normalizado que calcula a posição de um conceito na sua ontologia de referência. No seu valor mais baixo indica uma baixa especificidade e por isso um fraco nível de conhecimento. No seu valor mais alto reflete um alto nível de conhecimento, pois indica um conceito com especificidade máxima. A medida de cobertura de anotações indica o a razão entre o número de anotações (termos semânticos) com referência a conceitos de ontologias e o número total de anotações. Um baixo valor revela um número baixo de termos semânticos, para um maior número de anotações, o que indica uma baixa integração semântica dos metadados. Um valor alto, por seu turno, revela uma forte presença de termos semânticos em função das anotações existentes.

- **Plataforma de análise e avaliação de metadados**

  Foi desenvolvida uma plataforma que incorpora as medidas apresentadas por um estudo anterior e permite a avaliação da qualidade de integração semântica de metadados, de acordo com a sua integração com recursos externos das anotações utilizadas. A plataforma apresenta uma solução em camadas, totalmente distribuída, que tem na performance de análise e avaliação de metadados e na expansibilidade da sua estrutura a diferentes repositórios de dados científicos, dois dos seus requisitos de qualidade mais importantes. É composta por quatro camadas com uma alta coesão, i.e., com um conjunto de funcionalidades estritamente contidas dentro da fronteira lógica de cada uma, e com baixa interdependência, i.e., cada uma das camadas não necessita das restantes para satisfazer o seu papel. Numa representação vertical, a camada de topo é responsável por apresentar um interface a um utilizador, através do qual este pode submeter os metadados que pretende analisar. A camada imediatamente abaixo é responsável pelo suporte de um interface computacional, que permite estabelecer um ponto de contato através do qual outros meios computacionais podem interagir com a plataforma e submeter metadados para avaliação. O desenho de interoperabilidade que este interface oferece enquadra-se nos princípios de *Web* Semântica, possibilitando a integração dos seus recursos com outros domínios de conhecimento. Abaixo desta camada encontra-se o motor de análise e avaliação de metadados. Cabe-lhe desenvolver os esforços necessários de modo a encontrar, quer as anotações utilizadas na descrição dos metadados, quer os termos semânticos utilizados na integração a recursos externos, de modo a que possa ser calculado o valor de especificidade particular dos conceitos ontológicos utilizados, e global do ficheiro de metadados dado para avaliação. Na camada base de toda a arquitetura são salvaguardadas as ontologias necessárias para que o motor possa efetuar os cálculos necessários e transmitir uma medida de qualidade. Nesta camada as ontologias são representadas através de um modelo relacional, sobre o qual são executadas as fórmulas estudadas.

- **Avaliação da implementação da plataforma**

  Foi efetuada uma avaliação da implementação da plataforma através da sua aplicação a um repositório real de dados, o *Metabolights*, anotados semanticamente. Foram recolhidos os dados dessa aplicação e analisados de

acordo com os requisitos funcionais e não-funcionais, apresentados como objetivos da plataforma, assim como de acordo com uma análise quantitativa dos resultados obtidos. Foi também elaborada uma avaliação quantitativa do processo de cura, feito pela equipa do repositório, aos metadados de três estudos do repositório. Por fim foi colocado à disposição da equipa do *Metabolights* um questionário, de modo a aferir da usabilidade da solução apresentada como contribuição desta tese.

Os resultados obtidos nesta tese permitem chegar a algumas conclusões. É possível concluir que as medidas estudadas para a avaliação da qualidade de integração semântica de integração apresentam-se como válidas, pois traduzem resultados coerentes com aquilo que foi considerado expectável do ponto de vista do possível nível de conhecimento dado por uma anotação de termo. É também possível concluir que as avaliações efetuadas pelo motor de análise e avaliação são válidas, do ponto de vista da qualidade de integração semântica, pois estão de acordo, não apenas com os resultados de outros estudos anteriores sobre o tema, mas com uma verificação manual de resultados. Por fim, os resultados obtidos estão em consonância com o problema descrito nesta tese, e com aquilo que foi possível investigar no início do seu desenvolvimento. Estes revelam que, apesar da existência de algumas ferramentas tecnologicamente apropriadas para o efeito, o trabalho de integração semântica é feito sobretudo pelos investigadores [3], sendo que muitos não consideram a integração e partilha de dados como uma etapa essencial no sucesso e avanço da ciência.

Trabalho futuro

Foram estudadas várias ferramentas de anotação semântica, que tentam abstrair os investigadores da complexidade da criação, anotação, partilha e até gestão de metadados. No entanto, nenhuma estabelece um critério de avaliação da qualidade de integração semântica, do ponto de vista do conhecimento associado às descrições utilizadas para a caracterização dos conjuntos de dados produzidos durante as ações de investigação, i.e. não permitem aferir da potencial interoperabilidade proporcionada pelos metadados, tendo em conta a descoberta e compreensão dos dados quer por humanos, quer por meios computacionais.

Daquilo que foi possível investigar não existe ainda nenhuma ferramenta como aquela que foi construída nesta tese. Foi possível constatar esse facto através de uma apresentação da solução à equipa de desenvolvimento de uma das ferramentas estudadas, a plataforma ISA-TOOLS, descrita na Secção 2.4.2, desenvolvida pelo EMBL-EBI. Os resultados apresentados foram os suficientes para que a equipa ponderasse a integração da ferramenta desenvolvida nesta tese, na própria plataforma desenvolvida pelo EMBL-

EBI, recebendo inclusive por parte de membros do Oxford e-Research Centre[10] alguns conselhos sobre a forma como os resultados poderiam ser melhor apresentados na interface de utilizador.

Não apenas pelo estudo das medidas de avaliação da qualidade de integração semântica, que podem ser incorporadas em outras plataformas, ou em outros estudos, mas também pelos paradigmas de desenvolvimento utilizados na edificação da contribuição da tese, que permitem um ambiente de produção distribuído, com uma alta margem de modificabilidade, utilizada de modo singular por investigadores, através da interface de utilizador, ou utilizada de modo computacional, integrada com outras soluções através da interface computacional, ou até mesmo incorporada diretamente numa plataforma como a ISA-TOOLS, a solução encontrada pela contribuição desta tese pode ser ainda futuramente aprofundada para acomodar outros requisitos, de modo a ser colocada em produção para o apoio à criação de metadados de acordo com os princípios da *Web* Semântica.

[10] http://www.oerc.ox.ac.uk/about-us

Capítulo 6

Bibliografia

[1] F. M. Couto, "Rating, recognizing and rewarding metadata integration and sharing on the semantic web," *Proceeding URSW'14 Proceedings of the 10th International Conference on Uncertainty Reasoning for the Semantic Web,* vol. 1259, pp. 67-72, October 2014.

[2] G. Galilei, Sidereus Nuncius, or The Sidereal Messenger, University of Chicago Press, 1989.

[3] L. M. Federer, Y.-L. Lu, D. J. Joubert, J. Welsh e B. Brandys, "Biomedical Data Sharing and Reuse: Attitudes and Practices of Clinical and Scientific Research Staff," PLOS ONE Journal, June 2015.

[4] Innovative Medicines Initiative, "IMI Call for proposals on major challenges in drug development," April 2016. [Online]. Available: http://www.imi.europa.eu/sites/default/files/uploads/documents/IMI2Call9/IMI2_Call9_TopicsText.pdf. [Acedido em Abril 2016].

[5] M. Baker, "Quantitative data: learning to share," Nature Methods, vol. 9, p. 39–41, December 2012.

[6] B. Ballester, J. Baran, A. Cros, S. Durinck, H. Estrella, J. M. Guberman, S. Haider, R. Holland, J. Hsu, O. Kasprzyk, D. Keefe, Y. Liang, D. London, E. Marcora, C. Melsopp e L. Pandini, "BioMart: an open source data federation system and its applications to different types of biological data," 2011. [Online]. Available: http://www.biomart.org. [Acedido em Novembro 2015].

[7] "re3data: Registry of Research Data Repositories," 2012, [Online]. Available: http://service.re3data.org/about. [Acedido em Novembro 2015].

[8] E. K. Nelson, B. Piehler, J. Eckels, A. Rauch, M. Bellew, P. Hussey, S. Ramsay, C. Nathe, K. Lum, K. Krouse, D. Stearns, B. Connolly, T. Skillman e M. Igra, "LabKey Server: An open source platform for scientific data integration, analysis and collaboration," BMC Bioinformatics, March 2011.

[9] B. L. Millard, M. Niepel, M. P. Menden, J. L. Muhlich e P. K. Sorger, "Adaptive informatics for multifactorial and high-content biological data," Nature Methods, vol. 8, p. 487–492, 2011.

[10] G. Cheng, W. Ge e Y. Qu, "Falcons: searching and browsing entities on the semantic web," Proceedings of the 17th international conference on World Wide Web, pp. 1101-1102, 2008.

[11] M. d'Aquin e E. Motta, "Watson, more than a Semantic Web search engine," Journal Semantic Web, Vols. %1 de %22, Issue 1, pp. 55-63, January 2011.

[12] D. Dou, H. Wang e H. Liu, "Semantic data mining: A survey of ontology-based approaches," Semantic Computing (ICSC), 2015 IEEE International Conference on, pp. 244 - 251, february 2015.

[13] P. Sweeting, "Taking the Measure of Metadata," Media & Entertainment Services Alliance Journal, April 2015.

[14] T. Berners-Lee, J. Hendler e O. Lassilia, "The semantic web," Scientific American, vol. 284(5), p. 34–44, May 2001.

[15] K. Haug, R. M. Salek, P. Conesa, J. Hastings, P. de Matos, M. Rijnbeek, T. Mahendrakar, M. Williams, S. Neumann, P. Rocca-Serra, E. Maguire, A. González-Beltrán, S.-A. Sansone, J. L. Griffin e C. Steinbeck, "MetaboLights—an open-access general-purpose repository for metabolomics studies and associated meta-data," Nucleic Acids Research, January 2013.

[16] C. Ramos, M. Louro, M. Santos e F. M. Couto, "Knowledge Ratings in MetaboLights," 2015. [Online]. Available: arXiv:1604.07997v2 [cs.DL].

[17] T. Berners-Lee, "Linked Data - Design Issues," World Wide Web Consortium (W3C), 2006. [Online]. Available: http://www.w3.org/DesignIssues/LinkedData.html. [Acedido em Outubro 2015].

[18] C. Bizer, T. Heath e T. Berners-Lee, "Linked data - the story so far," International Journal on Semantic Web and Information Systems, vol. 5(3), p. 1–22, 2009.

[19] R. Cyganiak, D. Wood e M. Lanthaler, "RDF 1.1 Concepts and Abstract Syntax, W3C Recommendation," February 2014. [Online]. Available: https://www.w3.org/TR/2014/REC-rdf11-concepts-20140225. [Acedido em Dezembro 2015].

[20] T. Berners-Lee e J. Jaffe, "World Wide Web Consortium (W3C)," 1994, [Online]. Available: https://www.w3.org/Consortium. [Acedido em Outubro 2015].

[21] S. S. A. Harris, "SPARQL 1.1 Query Language - W3C Recommendation," March 2013. [Online]. Available: https://www.w3.org/TR/sparql11-query. [Acedido em Outubro 2015].

[22] T. Berners-Lee, R. Fielding e L. Masinter, "RFC 2396 - Uniform Resource Identifiers (URI): Generic Syntax," August 1998. [Online]. Available: http://www.isi.edu/in-notes/rfc2396.txt. [Acedido em Outubro 2015].

[23] V. G. Cerf e R. E. Kahn, "A Protocol for Packet Network Intercommunication," IEEE Transactions on Communications, Vols. %1 de %222, No. 5, p. 637–648, 1974.

[24] D. M. L. Brickley, "FOAF Vocabulary Specification 0.99 - Paddington Edition," January 2014. [Online]. Available: http://xmlns.com/foaf/spec/. [Acedido em Novembro 2015].

[25] D. Brickley, R. V. Guha e A. Layman, "RDF Schema 1.1, W3C Recommendation," February 2014. [Online]. Available: https://www.w3.org/TR/rdf-schema. [Acedido em Novembro 2015].

[26] S. Bechhofer, F. van Harmelen, J. Hendler, I. Horrocks, D. L. McGuinness, P. F. Patel-Schneider e L. A. Stein, "OWL Web Ontology Language, W3C Recommendation," February 2004. [Online]. Available: https://www.w3.org/TR/owl-ref. [Acedido em Novembro 2015].

[27] T. Berners-Lee, "The next web, TED Talk, TED.com," 2009. [Online]. Available: https://www.ted.com/talks/tim_berners_lee_on_the_next_web. [Acedido em Novembro 2015].

[28] "Linking Open Data Project," W3C SWEO Community Project, 2007. [Online]. Available: https://www.w3.org/wiki/SweoIG/TaskForces/CommunityProjects/LinkingOpenData. [Acedido em Janeiro 2016].

[29] "European Union Open Data Portal," [Online]. Available: https://open-data.europa.eu/en/about. [Acedido em January 2016].

[30] "O futuro do «Open e Linked Data» em debate no INE," Agência para a modernização administrativa, November 2015. [Online]. Available: http://www.dados.gov.pt/pt/noticias/o-futuro-do-%C2%ABopen-e-linked-

data%C2%BB-em-debate-no-ine-(1).aspx#sthash.ZxwEq93c.dpbs. [Acedido em Janeiro 2016].

[31] T. Heath, "Linked Data - Connect Distributed Data across the Web," Linked Data community, [Online]. Available: http://linkeddata.org/about. [Acedido em Novembro 2015].

[32] R. Cyganiak e A. Jentzsch, "The Linking Open Data cloud diagram," 2014. [Online]. Available: http://lod-cloud.net/versions/2014-08-30/lod-cloud.svg. [Acedido em Novembro 2015].

[33] M. Schmachtenberg, C. Bizer e H. Paulheim, "State of the LOD Cloud, Version 0.4," 8 August 2014. [Online]. Available: http://linkeddatacatalog.dws.informatik.uni-mannheim.de/state. [Acedido em Dezembro 2015].

[34] "DBpedia – A Large-scale, Multilingual Knowledge Base Extracted from Wikipedia," [Online]. Available: http://wiki.dbpedia.org/about. [Acedido em Janeiro 2016].

[35] S. Auer, C. Bizer, G. Kobilarov, J. Lehmann, R. Cyganiak e Z. Ives, "DBpedia: A Nucleus for a Web of Open Data," Proceeding ISWC'07/ASWC'07 Proceedings of the 6th international The semantic web and 2nd Asian conference on Asian semantic web conference, pp. 722-735, 2007.

[36] M. Uschold a e M. Gruningera, "Ontologies: principles, methods and applications," The Knowledge Engineering Review, Cambridge Journals, vol. 11, pp. 93-136, 1996.

[37] W3C OWL Working Group, "OWL 2 Web Ontology Language, W3C Recommendation," December 2012. [Online]. Available: https://www.w3.org/TR/owl-overview. [Acedido em Janeiro 2016].

[38] J. Richter, "The OBO Flat File Format Specification, version 1.2," May 2006. [Online]. Available: https://oboformat.googlecode.com/svn/trunk/doc/GO.format.obo-1_2.html. [Acedido em Março 2016].

[39] "Dryad Digital Repository," [Online]. Available: http://datadryad.org/pages/organization. [Acedido em Fevereiro 2016].

[40] "Biomedical sciences research infrastructures," [Online]. Available: http://www.biomedbridges.eu/biomedical-sciences-research-infrastructures. [Acedido em Fevereiro 2016].

[41] F. M. Couto, "KnowledgeCoin recognizing and rewarding metadata integration and sharing on the semantic web," Lightning Talk, ISWC2014, 2014.

[42] P. Rocca-Serra, S. Sansone e M. Brandizi, "Specification documentation: release candidate 1, ISA-TAB 1.0," November 2008. [Online]. Available: http://isatab.sourceforge.net/docs/ISA-TAB_release-candidate-1_v1.0_24nov08.pdf. [Acedido em Janeiro 2016].

[43] P. Rocca-Serra, M. Brandizi, E. Maguire, N. Sklyar, C. Taylor, K. Begley, D. Field, S. Harris, W. Hide, O. Hofmann, S. Neumann, P. Sterk, W. Tong e S.-A. Sansone, "ISA-TOOLS software suite: Supporting standards-compliant experimental annotation and enabling curation at the community level," Bioinformatics, Oxford Journals, vol. 26 (18), pp. 2354-2356, 2010.

[44] "BioPortal: Repository of biomedical ontologies," National Centers for Biomedical Computing, 2005. [Online]. Available: http://bioportal.bioontology.org. [Acedido em Novembro 2015].

[45] "ZOOMA," European Bioinformatics Institute, April 2016. [Online]. Available: http://www.ebi.ac.uk/spot/zooma/about.html. [Acedido em Abril 2016].

[46] J. Ferreira e F. M. Couto, "OWLtoSQL," 2012. [Online]. Available: https://github.com/jdferreira/OWLtoSQL. [Acedido em Outrubro 2015].

[47] H. Matthew e S. Bechhofer, "The OWL API: A Java API for OWL ontologies," Semantic Web, vol. 2(1), pp. 11-21, 2011.

[48] R. Fielding, "Architectural Styles and the Design of Network-based Software Architectures, Dissertion," 2000. [Online]. Available: https://www.ics.uci.edu/~fielding/pubs/dissertation/fielding_dissertation.pdf. [Acedido em Março 2016].

[49] Q. H. Mahmoud, "Sockets programming in Java: A tutorial," December 1996. [Online]. Available: http://www.javaworld.com/article/2077322/core-java/core-java-sockets-programming-in-java-a-tutorial.html. [Acedido em Fevereiro 2016].

[50] T. Berners-Lee, "Information Management: A Proposal, CERN," March 1990. [Online]. Available: https://www.w3.org/History/1989/proposal.html. [Acedido em Março 2016].

[51] jQuery Foundation, "jQuery," jQuery Foundation, [Online]. Available: https://jquery.com/. [Acedido em Fevereiro 2016].

[52] G. E. Krasner e S. T. Pope, "A cookbook for using the model–view controller user interface paradigm in Smalltalk-80," Journal of Object-Oriented Programming, Vols. %1 de %21, Issue 3, pp. 26 - 49, August 1988.

[53] A. Homer, "ASP.NET Patterns every developer should know," Junho 2011. [Online]. Available: http://www.developerfusion.com/article/8307/aspnet-patterns-every-developer-should-know. [Acedido em Dezembro 2015].

[54] The PHP Group, "PHP: Hypertext Preprocessor," The PHP Group, 2001. [Online]. Available: http://php.net/manual/en/intro-whatis.php. [Acedido em Dezembro 2015].

[55] Cake Software Foundation, "CakePHP: a web development framework," [Online]. Available: http://book.cakephp.org/3.0/en/index.html. [Acedido em Fevereiro 2016].

[56] D. Sheiko, "WebSockets vs Server-Sent Events vs Long-polling," May 2012. [Online]. Available: http://dsheiko.com/weblog/websockets-vs-sse-vs-long-polling. [Acedido em Dezembro 2015].

[57] M. Armbrust, A. Fox, R. Griffith, A. D. Joseph, R. H. Katz, A. Konwinski, G. Lee, D. A. Patterson, A. Rabkin, I. Stoica e M. Zaharia, "Above the Clouds: A Berkeley View of Cloud Computing, EECS Department, University of California, Berkeley," Magazine Communications of the ACM, Vols. %1 de %253, Issue 4, pp. 50-58, 2010.

[58] B. Butler, "PaaS Primer: What is platform as a service and why does it matter?," February 2013. [Online]. Available: http://www.networkworld.com/article/2163430/cloud-computing/paas-primer--what-is-platform-as-a-service-and-why-does-it-matter-.html. [Acedido em Março 2016].

[59] Red Hat, "OpenShift: Red Hat's Platform-as-a-Service (PaaS)," Red Hat, [Online]. Available: https://www.openshift.com. [Acedido em Janeiro 2016].

[60] Red Hat, "OpenShift Origin: The Open Source Application Container Platform," OpenShift, [Online]. Available: https://www.openshift.org. [Acedido em Janeiro 2016].

[61] Red Hat, "Introduction to Project Atomic," Red Hat, [Online]. Available: http://www.projectatomic.io/docs/introduction. [Acedido em Janeiro 2016].

[62] “What is Docker?,” Docker, 2014. [Online]. Available: https://www.docker.com/what-docker. [Acedido em Janeiro 2016].

[63] “Git --fast-version-control,” [Online]. Available: https://git-scm.com/about. [Acedido em Janeiro 2016].

[64] Apache Software Foundation, “Apache HTTP Server Project,” Apache Software Foundation, [Online]. Available: https://httpd.apache.org/ABOUT_APACHE.html. [Acedido em Janeiro 2016].

[65] “November 2015 Web Server Survey,” Netcraft, November 2015. [Online]. Available: http://news.netcraft.com/archives/2015/11/16/november-2015-web-server-survey.html. [Acedido em Março 2016].

[66] “The Apache Tomcat,” Apache Software Foundation, 1999. [Online]. Available: http://tomcat.apache.org. [Acedido em Fevereiro 2016].

[67] Oracle Corporation, “Jersey: RESTful Web Services in Java,” Oracle Corporation, 2010. [Online]. Available: https://jersey.java.net. [Acedido em Março 2016].

[68] Oracle Corporation, “Java API for RESTful Services (JAX-RS),” Java.net, [Online]. Available: https://jax-rs-spec.java.net. [Acedido em Feveiro 2016].

[69] E. F. Codd, “A Relational Model of Data for Large Shared Data Banks,” Communications of the ACM (Association for Computing Machinery), vol. 13 (6), p. 377–87, 1970.

[70] “ISO/IEC 9075-2:2011: Information technology -- Database languages -- SQL -- Part 2: Foundation (SQL/Foundation),” 2011. [Online]. Available: http://www.iso.org/iso/catalogue_detail.htm?csnumber=53682. [Acedido em Fevereiro 2016].

[71] R. T. Fielding, J. Gettys, J. C. Mogul, H. F. Nielsen, L. Masinter, P. J. Leach e T. Berners-Lee, “Hypertext Transfer Protocol -- HTTP/1.1,” IETF. RFC 2616, 1999.

[72] T. Berners-Lee, “The Original HTTP as defined in 1991,” 1991. [Online]. Available: https://www.w3.org/Protocols/HTTP/AsImplemented.html. [Acedido em Dezembro 2015].

[73] Y. Shafranovich, "RFC 4180 - Common Format and MIME Type for Comma-Separated Values (CSV) Files," October 2005. [Online]. Available: https://tools.ietf.org/html/rfc4180. [Acedido em January 2016].

[74] D. Crockford, "JSON: The Fat-Free Alternative to XML," December 2006. [Online]. Available: http://www.json.org/fatfree.html. [Acedido em Março 2016].

[75] L. Gorrie, A. Bengtsson, A. Ajad, R. DiFalco e B. Shanks, "Tuple Space," November 2014. [Online]. Available: http://c2.com/cgi/wiki?TupleSpace. [Acedido em Dezembro 2015].

[76] E. Gamma, J. Vlissides, R. Johnson e R. Helm, Design Patterns: Elements of Reusable Object-Oriented Software, Pearson Education, 1994.

[77] M. K. Houghton, "Tuplespace, a Java implementation of a Tuplespace," [Online]. Available: https://github.com/mike-k-houghton/tuplespace. [Acedido em Janeiro 2016].

[78] European Bioinformatics Institute (EMBL-EBI), "What is metabolomics?," [Online]. Available: https://www.ebi.ac.uk/training/online/course/introduction-metabolomics/what-metabolomics. [Acedido em Abril 2016].

[79] C. Golbreich, M. Horridge, I. Horrocks, B. Motik e R. Shearer, "OBO and OWL: leveraging semantic web technologies for the life sciences," Proceeding ISWC'07/ASWC'07 Proceedings of the 6th international The semantic web and 2nd Asian conference on Asian semantic web conference, pp. 169-182, 2007.

[80] R. Cyganiak, R. Delbru, H. Stenzhorn, G. Tummarello e S. Decker, "Semantic sitemaps: Efficient and flexible access to datasets on the semantic web," Proceedings of the 5th European Semantic Web Conference, 2008.

[81] D. Beckett e T. Berners-Lee, "Turtle - Terse RDF Triple Language, W3C Team Submission," March 2011. [Online]. Available: https://www.w3.org/TeamSubmission/turtle. [Acedido em Março 2016].

[82] M. A. Rodriguez e E. M. J., "Determining semantic similarity among entity classes from different ontologies," IEEE Transactions on Knowledge and Data Engineering, Vols. %1 de %215, Issue: 2, pp. 442 - 456, February 2003.

[83] T. B. C. Heath, Linked Data: Evolving the Web into a Global Data Space (1st edition). Synthesis Lectures on the Semantic Web: Theory and Technology, vol. 1:1, Morgan & Claypool, 2011, pp. 1-136.

[84] A. A. Alsheikh-Ali, W. Qureshi, M. H. Al-Mallah e J. P. A. Ioannidis, "Public Availability of Published Research Data in High-Impact Journals," PLoS ONE, vol. 6(9): e24357, September 2011.

[85] C. Machado, D. Rebholz-Schuhmann, A. Freitas e F. Couto, "The semantic web in translational medicine: current applications and future directions," Briefings in bioinformatics, vol. 16(1), pp. 89-103, January 2015.

[86] M. Bergman, "What is a reference concept?," December 2010. [Online]. Available: http://www.mkbergman.com/938/what-is-a-reference-concept. [Acedido em Janeiro 2016].

[87] N. Shadbolt, T. Berners-Lee e W. Hall, "The Semantic Web Revisited," IEEE Intelligent Systems, vol. 21, pp. 96-101, 2006.

[88] C. Bizer, T. Heath, T. Berners-Lee e M. Hausenblas, "4th linked data on the web workshop," Proceedings of the 20th international conference companion on World wide web, pp. 303-304, 2011.

[89] S. C. Kleene, "Representation of Events in Nerve Nets and Finite Automata," em Automata Studies, C. S. a. J. McCarthy, Ed., Princeton University Press, 1956, pp. 3-41.

Anexo A Vistas de arquitetura do MAA – Pacote Blackboard

Figura 6-1 - Vista dos principais pacotes do MAA.

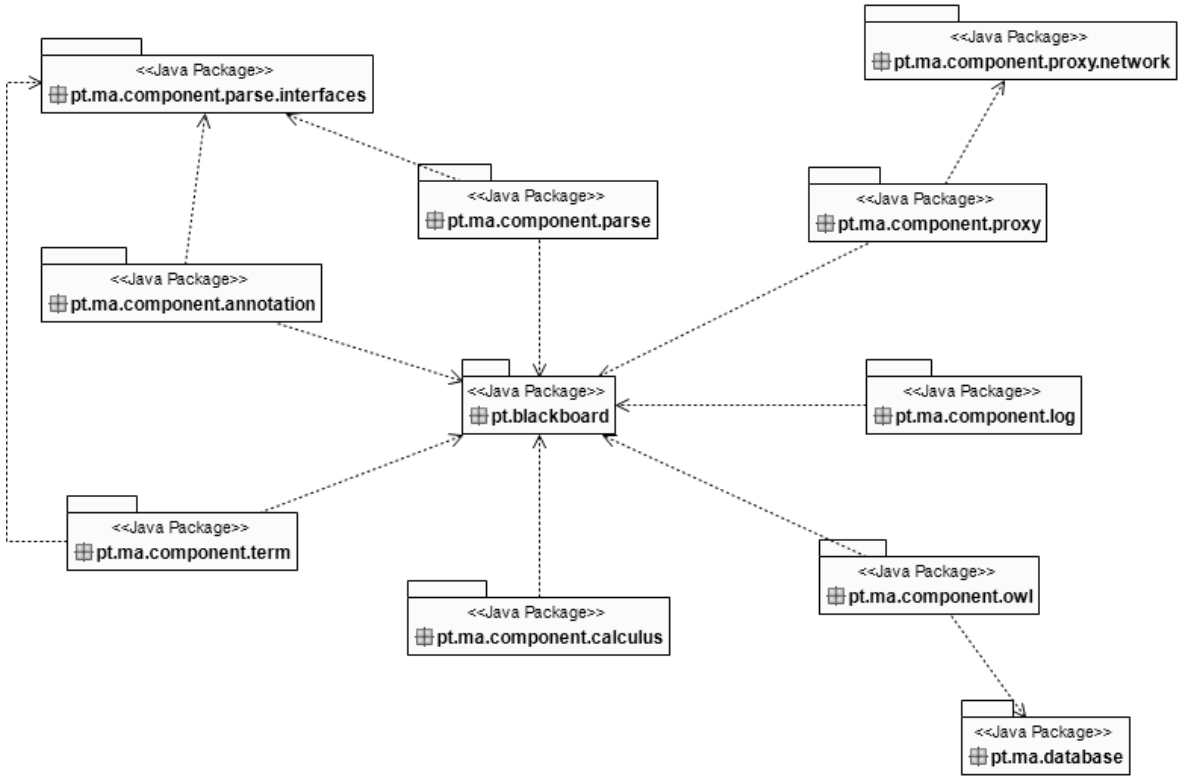
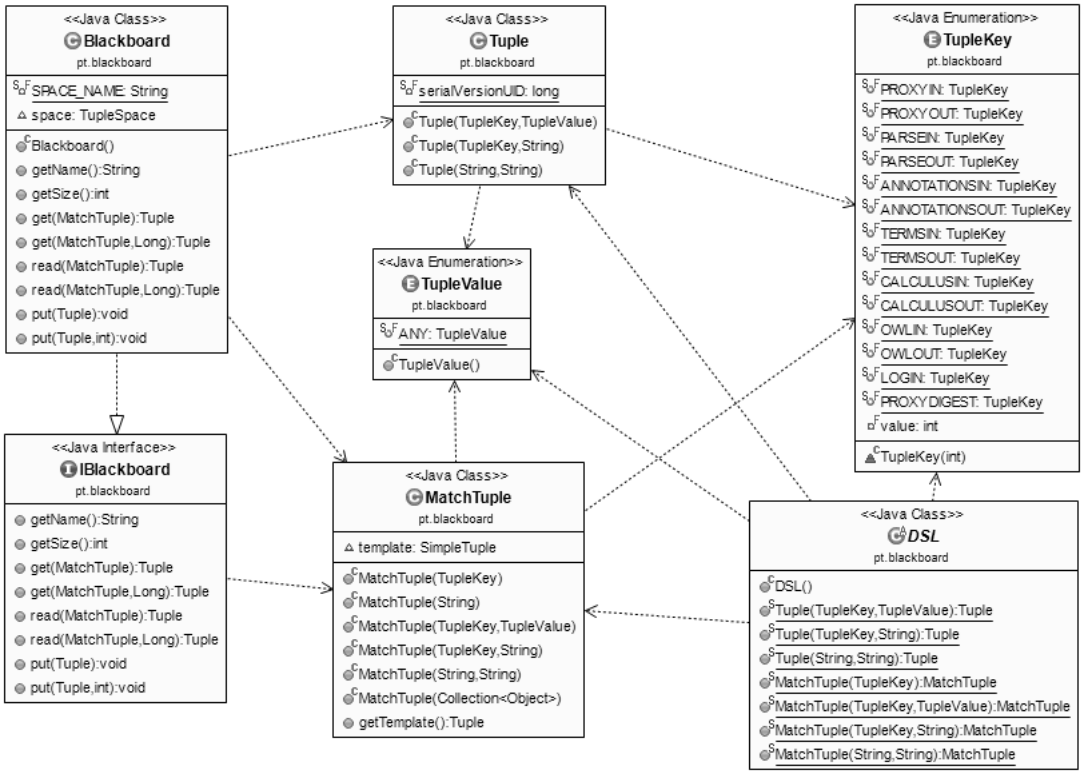


Figura 6-2 - Vista de classes do pacote pt.blackboard do MAA.



Anexo B Vistas de arquitetura do MAA – Pacote Proxy

Figura 6-3 - Vista de classes do pacote pt.ma.component.proxy do MAA.

Anexo C Vistas de arquitetura do MAA – Pacote Parser

Figura 6-4 - Vista de classes do pacote pt.ma.component.parser do MAA.

Anexo D Vistas de arquitetura do MAA – Pacote Term e Annotation

Figura 6-5 - Vista de classes do pacote pt.ma.component.term do MAA.

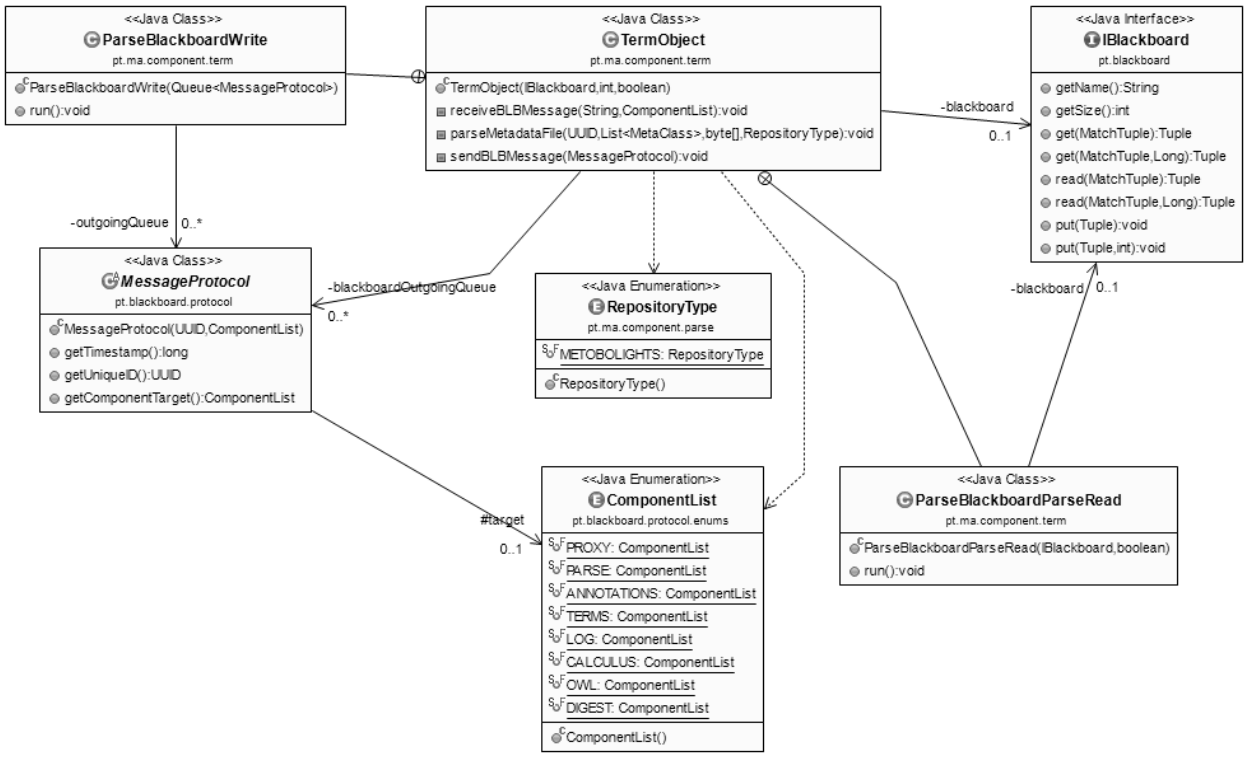
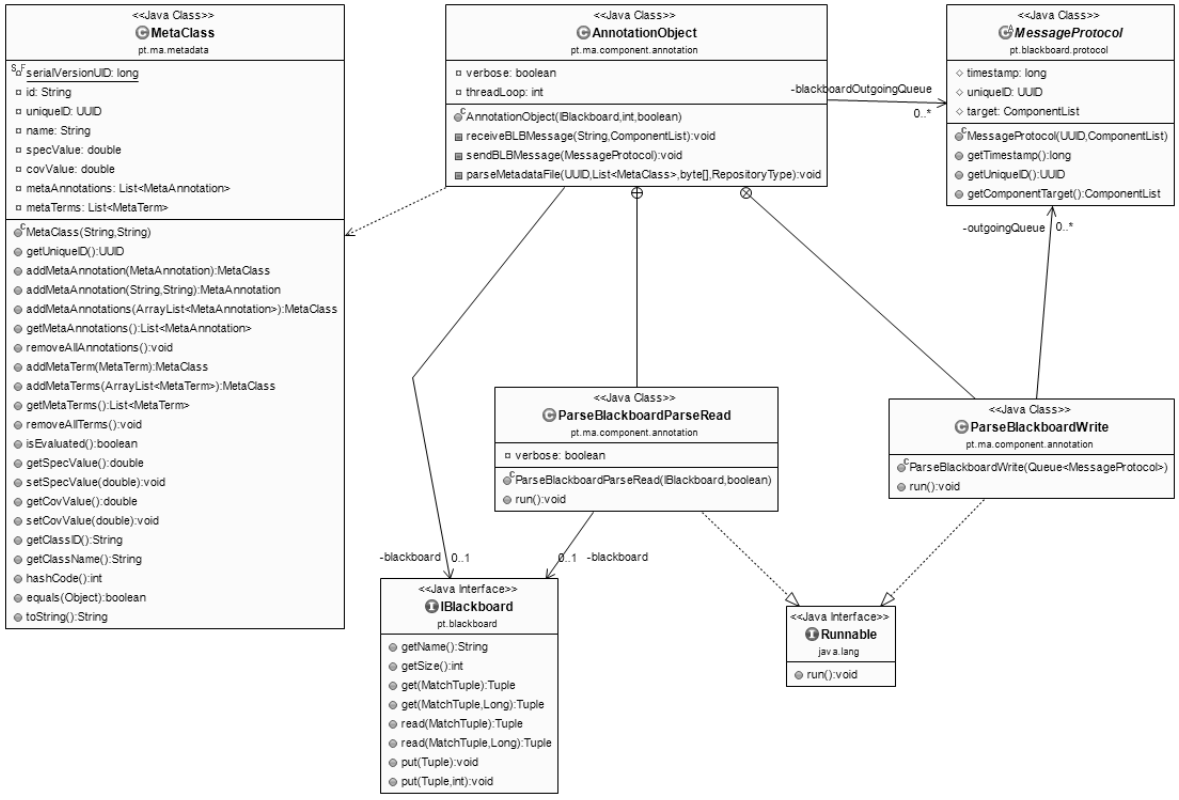


Figura 6-6 - Vista de classes do pacote pt.ma.component.annotation do MAA.



Anexo E Vistas de arquitetura do MAA – Pacote Calculus e OWL

Figura 6-7 - Vista de classes do pacote pt.ma.component.calculus do MAA.

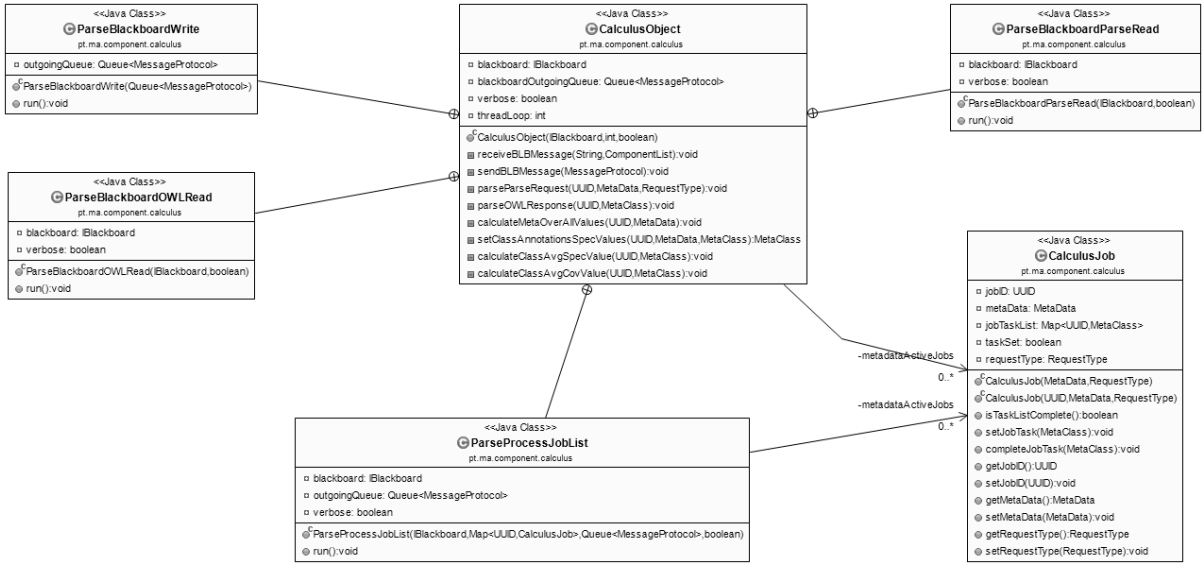
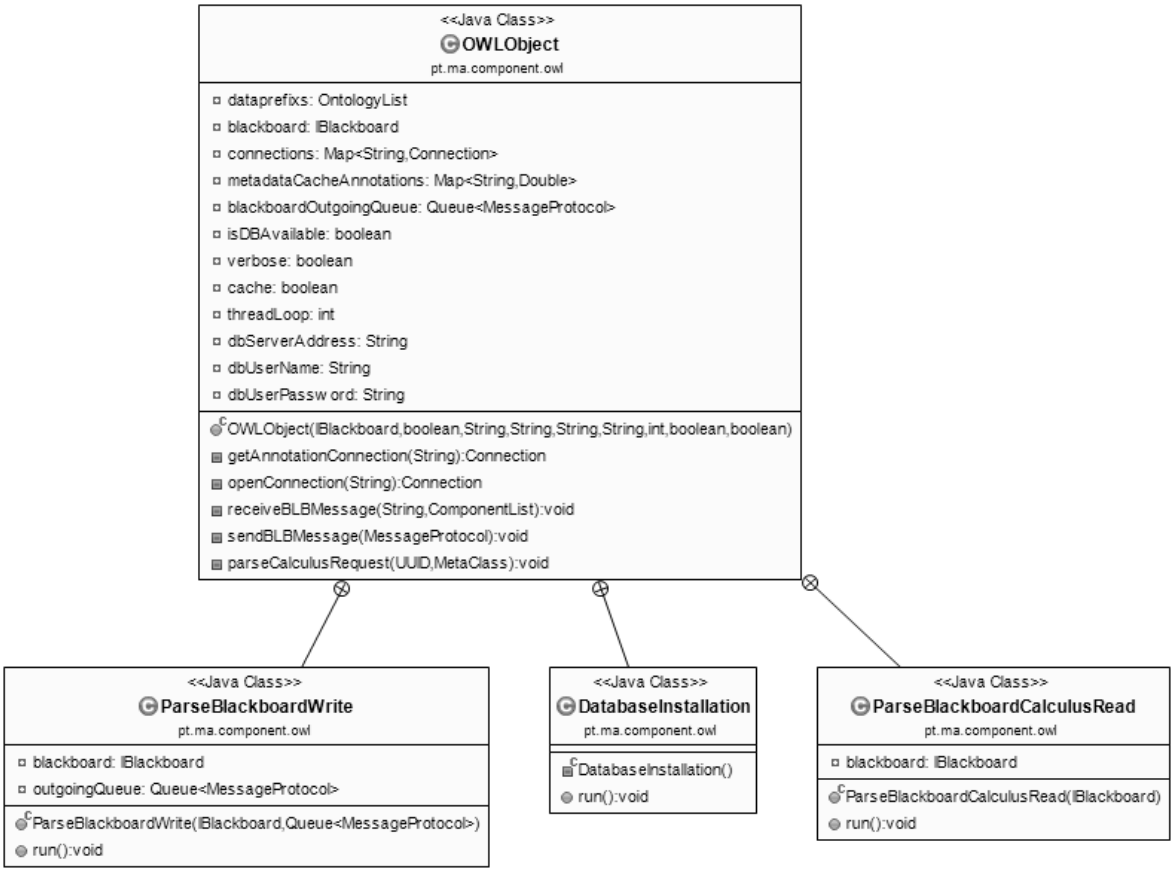
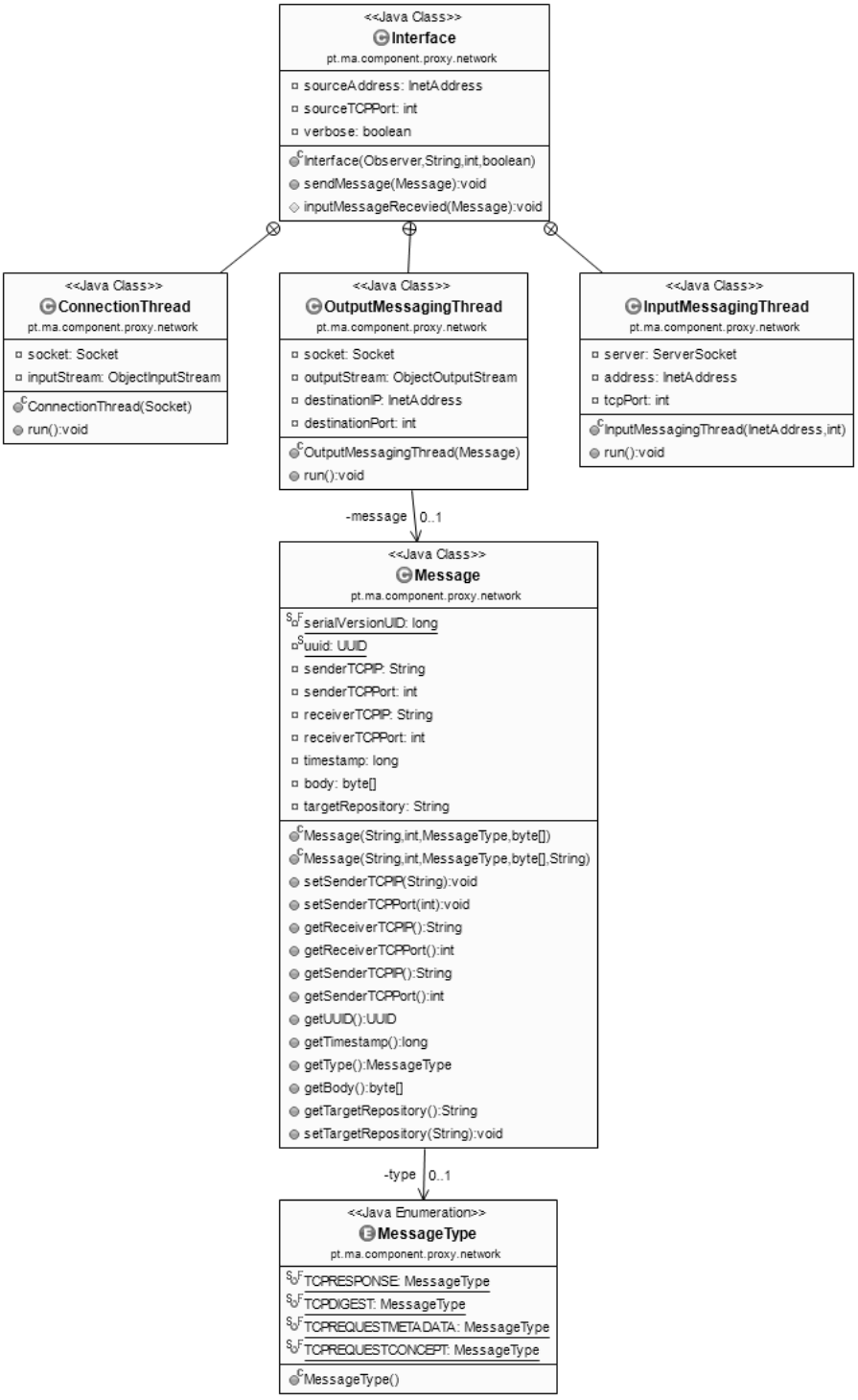


Figura 6-8 - Vista de classes do pacote pt.ma.component.owl do MAA.



Anexo F Vistas de arquitetura do MAA – Pacote Network

Figura 6-9 - Vista de classes do pacote pt.ma.component.proxy.network do MAA.



Anexo G Vistas de arquitetura do MAA – Pacote Log

Figura 6-10 - Vista de classes do pacote pt.ma.component.proxy.log do MAA.

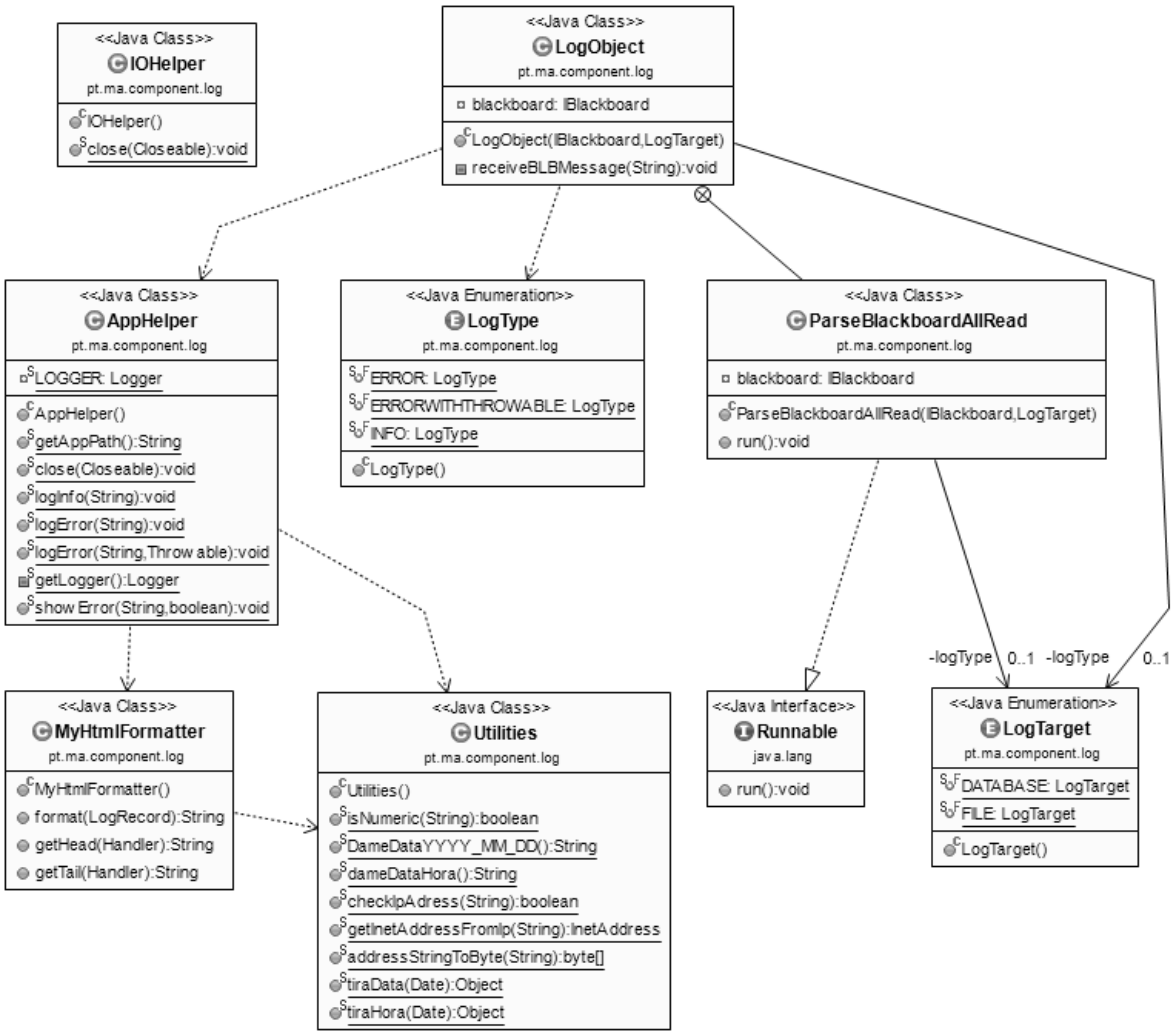


Tabela 6-1 - Lista de resultados para o procedimento de cálculo de especificidade.

		Class		Python Script					SQL				
Ontology	Ontology URI	Name	Id	Ancestors Count	Leaf Descendents Count	Execution Time (ms)	Specificity	Delta Time (ms)	Ancestors Count	Leaf Descendents Count	Execution Time (ms)	Specificity	Delta Spec
BIOMODELS (OWL)	http://data.bioontology.org/ontologies/BIOMODELS	epidermal initial	http://purl.org/obo/owl/CL#CL_0000538	8	1	3937	0,8889	3255	8	1	682	0,8889	0,0000
		meristematic cell	http://purl.org/obo/owl/CL#CL_0000685	5	4	8582	0,5556	8384	5	4	198	0,5556	0,0000
		plant cell	http://purl.org/obo/owl/CL#CL_0000610	4	36	78264	0,5180	78045	4	36	219	0,5180	0,0000
		smooth muscle cell	http://purl.org/obo/owl/CL#CL_0000192	12	6	14724	0,9231	14551	12	6	173	0,9231	0,0000
		muscle cell	http://purl.org/obo/owl/CL#CL_0000187	11	23	55083	0,7270	54923	11	23	160	0,7270	0,0000
		cell by organism	http://purl.org/obo/owl/CL#CL_0000004	2	776	2279942	0,1346	2279767	2	776	175	0,1346	0,0000
		Fascicular architecture of muscle	http://purl.org/obo/owlapi/fma#FMA_46569	5	37	62671	0,8186	62506	5	37	165	0,8186	0,0000
		Has_organizational pattern	http://purl.org/obo/owlapi/fma#FMA_9730	4	41	65740	0,6613	65600	4	41	140	0,6613	0,0000
		Anatomical relation	http://purl.org/obo/owlapi/fma#FMA_45723	2	171	339846	0,2803	339713	2	171	133	0,2803	0,0000
		Columnar cell	http://purl.org/obo/owlapi/fma#FMA_66771	2	4	9618	0,6667	9424	2	4	194	0,6667	0,0000
		Cell shape type	http://purl.org/obo/owlapi/fma#FMA_70632	1	8	16512	0,3636	16372	1	8	140	0,3636	0,0000
		Cell morphology	http://purl.org/obo/owlapi/fma#FMA_70631	1	14	23810	0,4375	23669	1	14	141	0,4375	0,0000
		differentiated	http://purl.org/obo/owl/PATO#PATO_0002099	4	3	8507	0,8000	8370	4	3	137	0,8000	0,0000
		cellular potency	http://purl.org/obo/owl/PATO#PATO_0001397	3	10	20524	0,6977	20360	3	10	164	0,6977	0,0000
		cellular quality	http://purl.org/obo/owl/PATO#PATO_0001396	2	35	65737	0,4046	65588	2	35	149	0,4046	0,0000
		blood vessel endothelial cell	http://purl.org/obo/owl/CL#CL_0000071	16	2	8347	0,9412	8085	16	2	262	0,9412	0,0000
		endothelial cell	http://purl.org/obo/owl/CL#CL_0000115	13	9	23253	0,8357	23097	13	9	156	0,8357	0,0000
		barrier cell	http://purl.org/obo/owl/CL#CL_0000215	4	14	36496	0,2569	36355	4	14	141	0,2569	0,0000
ICD10CM (TTL)	http://data.bioontology.org/ontologies/ICD10CM	Cerebral infarction due to thrombosis of cerebellar artery	http://purl.bioontology.org/ontology/ICD10CM/I63.34	4	3	5995	0,8000	3821	4	3	2174	0,8000	0,0000
		Cerebral infarction due to thrombosis of cerebral arteries	http://purl.bioontology.org/ontology/ICD10CM/I63.3	3	14	21913	0,6176	21563	3	14	350	0,6176	0,0000
		Cerebrovascular diseases (I60-I69)	http://purl.bioontology.org/ontology/ICD10CM/I60-I69	1	349	63499	0,2180	62973	1	349	526	0,2180	0,0000
		Fusion of spine	http://purl.bioontology.org/ontology/ICD10CM/M43.2	3	9	11978	0,7500	11646	3	9	332	0,7500	0,0000
		Other deforming dorsopathies	http://purl.bioontology.org/ontology/ICD10CM/M43	2	50	69632	0,4695	69292	2	50	340	0,4695	0,0000
		Chondropathies (M91-M94)	http://purl.bioontology.org/ontology/ICD10CM/M91-M94	1	179	328113	0,2157	327717	1	179	396	0,2157	0,0000
		Congenital iodine-deficiency syndrome	http://purl.bioontology.org/ontology/ICD10CM/E00	2	4	7378	0,6667	7049	2	4	329	0,6667	0,0000
		Diabetes mellitus (E08-E13)	http://purl.bioontology.org/ontology/ICD10CM/E08-E13	1	206	448036	0,2254	447625	1	206	411	0,2254	0,0000
		Malnutrition (E40-E46)	http://purl.bioontology.org/ontology/ICD10CM/E40-E46	1	8	16988	0,4444	16655	1	8	333	0,4444	0,0000
		Social phobias	http://purl.bioontology.org/ontology/ICD10CM/F40.1	3	2	8068	0,7500	7725	3	2	343	0,7500	0,0000
		Phobic anxiety disorders	http://purl.bioontology.org/ontology/ICD10CM/F40	2	23	45927	0,4340	45584	2	23	343	0,4340	0,0000
		Anxiety, dissociative, stress-related, somatoform and other	http://purl.bioontology.org/ontology/ICD10CM/F40-F48	1	66	104958	0,2619	104602	1	66	356	0,2619	0,0000
		Maternal care for disproportion due to hydrocephalic fetus	http://purl.bioontology.org/ontology/ICD10CM/O33.6	3	7	15647	0,7500	15315	3	7	332	0,7500	0,0000
		Maternal care for disproportion	http://purl.bioontology.org/ontology/ICD10CM/O33	2	34	52024	0,5231	51675	2	34	349	0,5231	0,0000
		Maternal care related to the fetus and amniotic cavity and p	http://purl.bioontology.org/ontology/ICD10CM/O30-O48	1	1122	1497133	0,1962	1496250	1	1122	883	0,1962	0,0000
		Abnormalities of heart beat	http://purl.bioontology.org/ontology/ICD10CM/R00	2	5	14644	0,6667	14286	2	5	358	0,6667	0,0000
		Abnormal blood-pressure reading, without diagnosis	http://purl.bioontology.org/ontology/ICD10CM/R03	2	2	9754	0,6667	9424	2	2	330	0,6667	0,0000
		Symptoms and signs involving the circulatory and respirator	http://purl.bioontology.org/ontology/ICD10CM/R00-R09	1	48	76232	0,2981	75888	1	48	344	0,2981	0,0000
SNMI (TTL)	http://data.bioontology.org/ontologies/SNMI	Cystic disease, NOS	http://purl.bioontology.org/ontology/SNMI/M-74300	3	3	4866	0,7500	3982	3	3	884	0,7500	0,0000
		Adenosis, NOS	http://purl.bioontology.org/ontology/SNMI/M-74200	3	4	5976	0,7500	5520	3	4	456	0,7500	0,0000
		EPITHELIAL DYSPLASIAS	http://purl.bioontology.org/ontology/SNMI/M-74400	3	8	10677	0,7500	10208	3	8	469	0,7500	0,0000
		Dysplasia, NOS	http://purl.bioontology.org/ontology/SNMI/M-74000	2	26	42102	0,5361	41672	2	26	430	0,5361	0,0000
		Metaplasia, NOS	http://purl.bioontology.org/ontology/SNMI/M-73000	2	23	37169	0,5476	36645	2	23	524	0,5476	0,0000
		Involution, NOS	http://purl.bioontology.org/ontology/SNMI/M-70300	2	3	6456	0,6667	5950	2	3	506	0,6667	0,0000
		Bacterin - toxoid, NOS	http://purl.bioontology.org/ontology/SNMI/C-E3000	2	62	79116	0,6667	78250	2	62	866	0,6667	0,0000
		Bacterin, NOS	http://purl.bioontology.org/ontology/SNMI/C-E1000	2	96	128887	0,6667	128173	2	96	714	0,6667	0,0000
		Vaccine - bacterin - toxoid, NOS	http://purl.bioontology.org/ontology/SNMI/C-E2000	2	114	151594	0,6667	150857	2	114	737	0,6667	0,0000
		Endoscope, NOS	http://purl.bioontology.org/ontology/SNMI/A-28000	3	22	30160	0,7500	29469	3	22	691	0,7500	0,0000
		Cannula, NOS	http://purl.bioontology.org/ontology/SNMI/A-26000	3	43	65029	0,7500	64575	3	43	453	0,7500	0,0000
		Adhesive, NOS	http://purl.bioontology.org/ontology/SNMI/A-13000	3	22	38240	0,7500	37839	3	22	401	0,7500	0,0000
OMIM (TTL)	http://data.bioontology.org/ontologies/OMIM	Lung	http://purl.bioontology.org/ontology/OMIM/MTHU000106	1	266	299474	0,4716	297778	1	266	1696	0,4716	0,0000
		Larynx	http://purl.bioontology.org/ontology/OMIM/MTHU001728	1	70	78131	0,4516	77838	1	70	293	0,4516	0,0000
		Airways	http://purl.bioontology.org/ontology/OMIM/MTHU001730	1	75	80950	0,4360	80640	1	75	310	0,4360	0,0000
		Ureters	http://purl.bioontology.org/ontology/OMIM/MTHU001614	1	47	51839	0,4273	51573	1	47	266	0,4273	0,0000
		External genitalia, female	http://purl.bioontology.org/ontology/OMIM/MTHU001848	1	83	74099	0,4089	73818	1	83	281	0,4089	0,0000
		Bladder	http://purl.bioontology.org/ontology/OMIM/MTHU000477	1	95	80530	0,4481	80246	1	95	284	0,4481	0,0000
		Face	http://purl.bioontology.org/ontology/OMIM/MTHU000089	1	619	708404	0,4269	707982	1	619	422	0,4269	0,0000
		External genitalia, male	http://purl.bioontology.org/ontology/OMIM/MTHU000063	1	129	198829	0,4188	198548	1	129	281	0,4188	0,0000
		Ears	http://purl.bioontology.org/ontology/OMIM/MTHU000052	1	706	794185	0,4886	793732	1	706	453	0,4886	0,0000
		Gastrointestinal	http://purl.bioontology.org/ontology/OMIM/MTHU000224	1	575	804555	0,4648	804160	1	575	395	0,4648	0,0000
		Biliary tract	http://purl.bioontology.org/ontology/OMIM/MTHU003422	1	46	58237	0,4510	57956	1	46	281	0,4510	0,0000
		Pancreas	http://purl.bioontology.org/ontology/OMIM/MTHU000376	1	61	92301	0,4841	92019	1	61	282	0,4841	0,0000
VTO (OWL)	http://data.bioontology.org/ontologies/VTO	Cnemidocarpa	http://purl.obolibrary.org/obo/VTO_0059139	5	3	5824	0,8333	5327	5	3	497	0,8333	0,0000
		Styelidae	http://purl.obolibrary.org/obo/VTO_0059078	4	49	70160	0,6667	70109	4	49	51	0,6667	0,0000
		Ascidiacea	http://purl.obolibrary.org/obo/VTO_0058822	2	265	373419	0,3046	373402	2	265	17	0,3046	0,0000
		Climatiformes	http://purl.obolibrary.org/obo/VTO_9011048	6	26	45544	0,7358	45529	6	26	15	0,7358	0,0000
		Culmacanthidae	http://purl.obolibrary.org/obo/VTO_9026262	7	1	4147	0,7778	4133	7	1	14	0,7778	0,0000
		Culmacanthus	http://purl.obolibrary.org/obo/VTO_9026260	8	1	3626	0,8889	3610	8	1	16	0,8889	0,0000
		Neomyxine	http://purl.obolibrary.org/obo/VTO_9046416	7	1	11284	0,8750	11272	7	1	12	0,8750	0,0000
		Myxinidae	http://purl.obolibrary.org/obo/VTO_0058702	6	73	126591	0,7526	126572	6	73	19	0,7526	0,0000
		Myxinomorphi	http://purl.obolibrary.org/obo/VTO_9002086	3	74	115330	0,3769	115317	3	74	13	0,3769	0,0000
		Hertzina	http://purl.obolibrary.org/obo/VTO_9024152	4	4	12972	0,8000	12959	4	4	13	0,8000	0,0000
		Conodonta	http://purl.obolibrary.org/obo/VTO_9024151	3	411	697458	0,5194	697445	3	411	13	0,5194	0,0000
		Archeognathus	http://purl.obolibrary.org/obo/VTO_9026555	4	2	4487	0,8000	4472	4	2	15	0,8000	0,0000

Anexo I Lista de estudos relevantes para teste do MAA de metadados

Tabela 6-3 - Lista de valores médios para os estudos do Metabolights.

	Analysis Rating Engine					Python Procedure					Manual Procedure				
Study ID	Total Annot.	Found Annot.	Terms	Avg. Specific.	Avg. Cover.	Total Annot.	Found Annot.	Terms	Avg. Specific.	Avg. Cover.	Total Annot.	Found Annot.	Terms	Avg. Specific.	Avg. Cover.
MTBLS1	9	6	20	0,8884	0,3000	0	0	0	0,0000	0,0000	9	6	20	0,8900	0,3000
MTBLS36	3	3	17	0,9667	0,1765	0	0	0	0,0000	0,0000	3	3	17	0,9667	0,1765
MTBLS88	5	5	16	0,7568	0,3125	7	7	16	0,6981	0,7500	5	5	16	0,7569	0,3125
MTBLS110	4	2	14	0,9167	0,1429	4	4	14	0,8787	0,5000	4	4	14	0,8421	0,2857
MTBLS137	4	3	15	0,9444	0,2000	3	3	15	0,8735	0,3750	4	3	15	0,9443	0,2000
MTBLS166	5	3	21	1,0000	0,1429	5	5	21	0,0000	0,5416	5	5	21	0,6000	0,2381
Avg:	5,00	3,67	17,17	0,9122	0,2124	3,17	3,17	11,00	0,4084	0,3611	5,00	4,33	17,17	0,8333	0,2521
Sum:	30	22	103			19	19	66			30	26	103		

Tabela 6-4 - Lista de estudos relevantes do repositório Metoboligths.

MTBLS1			
URI	ARE Specific.	Python Specific.	Manual Specific.
http://www.ebi.ac.uk/efo/EFO_0000400	0,7500	0,0000	0,7500
http://www.ebi.ac.uk/efo/EFO_0000195	1,0000	0,0000	1,0000
http://purl.obolibrary.org/obo/CHMO_0000591	0,5800	0,0000	0,5899
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C70665	-1,0000	0,0000	-1,0000
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C17357	-1,0000	0,0000	-1,0000
http://www.ebi.ac.uk/efo/EFO_0000195	1,0000	0,0000	1,0000
http://purl.obolibrary.org/obo/OBI_0000366	1,0000	0,0000	1,0000
http://purl.obolibrary.org/obo/OBI_0000623	1,0000	0,0000	1,0000
http://www.acdlabs.com/products/adh/nmr/1d_man/	-1,0000	0,0000	-1,0000
MTBLS36	ARE Specific.	Python Specific.	Manual Specific.
http://purl.obolibrary.org/obo/UO_0000033	1,0000	0,0000	1,0000
http://purl.obolibrary.org/obo/OBI_0000366	1,0000	0,0000	1,0000
http://purl.obolibrary.org/obo/OBI_0001478	0,9000	0,0000	0,9000
MTBLS88	ARE Specific.	Python Specific.	Manual Specific.
http://purl.obolibrary.org/obo/CHMO_0000575	0,6560	0,6557	0,6557
http://purl.obolibrary.org/obo/CHMO_0000796	0,5810	0,5810	0,5810
http://purl.obolibrary.org/obo/OBI_0000747	0,7140	0,7500	0,7143
http://purl.obolibrary.org/obo/OBI_0000366	1,0000	1,0000	1,0000
http://purl.obolibrary.org/obo/OBI_0000470	0,8330	0,8571	0,8333
MTBLS110	ARE Specific.	Python Specific.	Manual Specific.
http://purl.bioontology.org/ontology/MESH/D010937	-1,0000	1,0000	1,0000
http://purl.bioontology.org/ontology/MESH/D002338	-1,0000	0,8755	0,5352
http://purl.obolibrary.org/obo/OBI_0000366	1,0000	1,0000	1,0000
http://purl.obolibrary.org/obo/OBI_0000470	0,8330	0,8571	0,8333
MTBLS137	ARE Specific.	Python Specific.	Manual Specific.
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C62709	-1,0000	-1,0000	-1,0000
http://purl.obolibrary.org/obo/OBI_0001627	1,0000	1,0000	1,0000
http://purl.obolibrary.org/obo/OBI_0000366	1,0000	1,0000	1,0000
http://purl.obolibrary.org/obo/OBI_0000470	0,8330	0,8571	0,8330
MTBLS166	ARE Specific.	Python Specific.	Manual Specific.
http://purl.bioontology.org/ontology/MESH/C089796	-1,0000	0,0000	0,0000
http://purl.obolibrary.org/obo/OBI_0000366	1,0000	1,0000	1,0000
http://purl.bioontology.org/ontology/MESH/C089796	-1,0000	0,0000	0,0000
http://purl.obolibrary.org/obo/OBI_0000366	1,0000	1,0000	1,0000
http://purl.obolibrary.org/obo/OBI_0000623	1,0000	-1,0000	1,0000

Anexo J Resultados do MAA de metadados - Especificidade

Tabela 6-5 - Lista de resultados de especificidade por estudo do MAA.

	Design Class			Factor Class			Assay Class			Protocol Class			Overall		
Study ID	Total Anno.	Found Anno.	Specificity	Total Anno.	Found Anno.	Specificity	Total Anno.	Found Anno.	Specificity	Total Anno.	Found Anno.	Specificity	Avg. Specificity	Total Anno.	Found Anno.
MTBLS1	4	3	0,777	2	1	1,000	2	2	1,000	1	0	0,000	0,888	9	6
MTBLS10	4	4	0,895	2	2	0,482	2	2	0,917	0	0	0,000	0,797	8	8
MTBLS100	2	1	0,000	0	0	0,000	2	2	1,000	0	0	0,000	0,667	4	3
MTBLS102	6	3	0,789	0	0	0,000	2	2	1,000	0	0	0,000	0,874	8	5
MTBLS103	0	0	0,000	1	0	0,000	3	3	0,944	0	0	0,000	0,944	4	3
MTBLS104	1	0	0,000	0	0	0,000	2	2	1,000	0	0	0,000	1,000	3	2
MTBLS105	2	2	0,879	1	1	0,228	2	2	0,917	0	0	0,000	0,764	5	5
MTBLS107	0	0	0,000	2	0	0,000	2	2	0,917	0	0	0,000	0,917	4	2
MTBLS108	0	0	0,000	2	0	0,000	2	2	0,917	0	0	0,000	0,917	4	2
MTBLS109	0	0	0,000	2	0	0,000	2	2	0,917	0	0	0,000	0,917	4	2
MTBLS11	4	2	0,824	3	2	0,851	2	2	0,917	2	0	0,000	0,864	11	6
MTBLS110	0	0	0,000	2	0	0,000	2	2	0,917	0	0	0,000	0,917	4	2
MTBLS111	0	0	0,000	2	0	0,000	2	2	0,917	0	0	0,000	0,917	4	2
MTBLS112	2	1	1,000	3	1	0,750	2	2	0,917	0	0	0,000	0,896	7	4
MTBLS113	5	2	1,000	6	3	1,000	2	2	0,917	0	0	0,000	0,976	13	7
MTBLS114	1	0	0,000	1	0	0,000	2	2	1,000	0	0	0,000	1,000	4	2
MTBLS116	5	5	0,916	2	0	0,000	2	2	1,000	0	0	0,000	0,940	9	7
MTBLS117	1	1	0,656	3	3	0,798	2	2	0,917	0	0	0,000	0,814	6	6
MTBLS118	1	1	0,648	3	3	0,798	2	2	0,917	0	0	0,000	0,812	6	6
MTBLS119	3	2	0,938	2	1	0,800	2	2	0,917	0	0	0,000	0,902	7	5
MTBLS12	4	2	0,824	3	2	0,851	2	2	0,917	1	0	0,000	0,864	10	6
MTBLS120	3	2	0,938	1	1	0,800	2	2	0,917	0	0	0,000	0,902	6	5
MTBLS123	3	2	0,759	3	3	0,806	2	2	1,000	0	0	0,000	0,848	8	7
MTBLS124	1	0	0,000	2	0	0,000	2	2	0,917	0	0	0,000	0,917	5	2
MTBLS125	4	3	0,708	1	1	0,605	2	2	0,917	0	0	0,000	0,761	7	6
MTBLS126	3	2	0,618	2	1	1,000	2	2	0,917	0	0	0,000	0,814	7	5
MTBLS127	3	2	0,512	2	2	0,587	2	2	0,917	0	0	0,000	0,672	7	6
MTBLS128	5	2	0,750	2	2	0,821	2	2	0,950	0	0	0,000	0,840	9	6
MTBLS13	4	2	0,824	3	2	0,851	2	2	0,917	1	0	0,000	0,864	10	6
MTBLS131	1	1	1,000	2	2	0,833	2	2	1,000	0	0	0,000	0,933	5	5
MTBLS132	1	1	1,000	2	2	0,833	2	2	1,000	0	0	0,000	0,933	5	5
MTBLS133	1	1	1,000	0	0	0,000	2	2	1,000	0	0	0,000	1,000	3	3
MTBLS134	1	1	1,000	0	0	0,000	2	2	1,000	0	0	0,000	1,000	3	3
MTBLS137	0	0	0,000	2	1	1,000	2	2	0,917	0	0	0,000	0,944	4	3
MTBLS14	4	2	0,414	6	3	0,901	2	2	0,917	1	0	0,000	0,766	13	7
MTBLS140	0	0	0,000	3	3	0,917	2	2	0,917	0	0	0,000	0,917	5	5
MTBLS143	7	1	1,000	2	2	0,802	2	2	0,917	0	0	0,000	0,888	11	5
MTBLS144	4	2	0,693	2	2	0,875	2	2	0,917	0	0	0,000	0,828	8	6
MTBLS146	3	3	0,504	4	2	1,000	2	2	0,917	0	0	0,000	0,763	9	7
MTBLS147	2	2	0,410	0	0	0,000	2	2	1,000	1	1	0,000	0,564	5	5
MTBLS148	3	2	0,755	6	6	0,565	2	2	0,917	0	0	0,000	0,674	11	10
MTBLS15	4	2	0,414	6	3	0,901	2	2	0,917	1	0	0,000	0,766	13	7
MTBLS150	3	1	1,000	3	2	0,505	2	2	0,917	0	0	0,000	0,769	8	5
MTBLS152	3	1	1,000	2	2	0,505	2	2	0,917	0	0	0,000	0,769	7	5
MTBLS154	4	2	0,693	3	3	0,852	2	2	0,917	0	0	0,000	0,825	9	7
MTBLS155	4	2	0,693	2	2	0,875	2	2	0,917	0	0	0,000	0,828	8	6
MTBLS156	0	0	0,000	0	0	0,000	2	2	1,000	0	0	0,000	1,000	2	2
MTBLS157	3	2	0,693	3	3	0,917	2	2	0,917	0	0	0,000	0,853	8	7
MTBLS16	4	2	0,414	6	3	0,901	2	2	0,917	1	0	0,000	0,766	13	7
MTBLS161	7	4	0,645	2	1	1,000	2	2	1,000	0	0	0,000	0,797	11	7
MTBLS162	3	1	0,750	4	3	0,799	2	2	0,917	0	0	0,000	0,830	9	6
MTBLS163	5	1	0,700	0	0	0,000	2	2	0,917	0	0	0,000	0,844	7	3
MTBLS165	2	1	0,581	3	2	0,775	2	2	0,917	0	0	0,000	0,793	7	5
MTBLS166	2	1	1,000	1	0	0,000	2	2	1,000	0	0	0,000	1,000	5	3
MTBLS168	0	0	0,000	1	1	0,643	2	2	0,917	0	0	0,000	0,825	3	3
MTBLS169	3	3	0,917	1	1	0,800	2	2	0,950	0	0	0,000	0,908	6	6
MTBLS17	3	3	0,920	3	3	0,743	2	2	0,917	0	0	0,000	0,852	8	8
MTBLS170	2	2	0,917	1	1	0,326	2	2	0,917	0	0	0,000	0,798	5	5
MTBLS171	3	3	0,549	2	1	0,750	2	2	0,917	0	0	0,000	0,705	7	6
MTBLS172	3	2	0,679	0	0	0,000	2	2	1,000	0	0	0,000	0,840	5	4
MTBLS173	4	4	0,810	3	3	0,811	2	2	0,917	0	0	0,000	0,834	9	9
MTBLS174	4	2	0,790	5	4	0,877	2	2	1,000	0	0	0,000	0,886	11	8
MTBLS175	3	1	0,648	0	0	0,000	2	2	0,917	0	0	0,000	0,827	5	3
MTBLS176	2	2	1,000	5	0	0,000	2	2	0,917	0	0	0,000	0,958	9	4
MTBLS177	6	3	0,780	3	2	1,000	2	2	1,000	0	0	0,000	0,906	11	7
MTBLS178	2	1	1,000	0	0	0,000	2	2	0,917	0	0	0,000	0,944	4	3
MTBLS187	6	2	0,753	3	3	1,000	2	2	0,917	0	0	0,000	0,906	11	7
MTBLS188	2	2	1,000	2	2	0,703	2	2	0,917	0	0	0,000	0,873	6	6
MTBLS19	3	3	0,920	3	3	0,743	2	2	0,917	0	0	0,000	0,852	8	8
MTBLS191	4	3	0,549	2	2	1,000	2	2	0,917	0	0	0,000	0,783	8	7
MTBLS194	0	0	0,000	0	0	0,000	2	2	0,917	0	0	0,000	0,917	2	2
MTBLS197	8	3	0,455	1	1	0,800	2	2	0,917	0	0	0,000	0,666	11	6
MTBLS2	1	1	1,000	2	2	0,900	2	2	0,950	0	0	0,000	0,940	5	5
MTBLS20	3	3	0,885	1	1	0,714	2	2	0,917	2	0	0,000	0,867	8	6
MTBLS202	3	3	1,000	1	1	1,000	2	2	0,950	0	0	0,000	0,983	6	6
MTBLS203	5	2	1,000	3	1	1,000	2	2	0,917	0	0	0,000	0,967	10	5
MTBLS208	3	2	0,824	1	1	0,750	2	2	0,917	0	0	0,000	0,846	6	5
MTBLS21	3	2	0,824	2	1	0,807	2	2	0,917	0	0	0,000	0,858	7	5
MTBLS210	3	2	0,728	1	1	1,000	2	2	0,917	1	0	0,000	0,858	7	5
MTBLS212	5	4	0,888	2	1	1,000	2	2	0,917	0	0	0,000	0,912	9	7
MTBLS213	2	1	1,000	2	1	0,807	2	2	0,917	0	0	0,000	0,910	6	4
MTBLS214	2	1	0,121	2	2	1,000	2	2	0,917	0	0	0,000	0,791	6	5
MTBLS215	2	2	0,657	1	1	0,000	2	2	0,917	0	0	0,000	0,630	5	5
MTBLS218	3	2	1,000	2	1	1,000	2	2	0,917	0	0	0,000	0,967	7	5
MTBLS219	3	2	0,741	2	1	1,000	2	2	0,917	0	0	0,000	0,863	7	5
MTBLS22	4	2	0,414	3	1	0,702	2	2	0,917	1	0	0,000	0,673	10	5
MTBLS226	4	3	0,710	3	2	0,875	2	2	0,917	0	0	0,000	0,816	9	7
MTBLS228	1	0	0,000	1	1	1,000	2	2	0,917	0	0	0,000	0,944	4	3
MTBLS229	1	0	0,000	1	1	1,000	2	2	0,917	0	0	0,000	0,944	4	3
MTBLS23	1	1	0,648	0	0	0,000	2	2	0,917	0	0	0,000	0,827	3	3
MTBLS234	2	2	1,000	2	2	0,903	2	2	0,917	0	0	0,000	0,940	6	6
MTBLS235	1	1	1,000	0	0	0,000	2	2	0,917	0	0	0,000	0,944	3	3
MTBLS24	2	1	0,580	2	2	0,875	2	2	1,000	1	0	0,000	0,866	7	5
MTBLS243	5	5	0,602	1	0	0,000	2	2	0,917	1	0	0,000	0,692	9	7
MTBLS25	2	1	0,580	0	0	0,000	2	2	1,000	2	0	0,000	0,860	6	3
MTBLS26	6	3	0,885	2	1	0,800	2	2	0,950	0	0	0,000	0,893	10	6
MTBLS263	3	2	0,618	2	2	1,000	2	2	0,917	0	0	0,000	0,845	7	6
MTBLS264	3	2	0,618	3	1	0,750	2	2	0,917	0	0	0,000	0,764	8	5

	Design Class			Factor Class			Assay Class			Protocol Class			Overall		
Study ID	Total Anno.	Found Anno.	Specificity	Total Anno.	Found Anno.	Specificity	Total Anno.	Found Anno.	Specificity	Total Anno.	Found Anno.	Specificity	Avg. Specificity	Total Anno.	Found Anno.
MTBLS265	2	2	0,618	2	0	0,000	2	2	0,917	0	0	0,000	0,767	6	4
MTBLS266	2	2	0,618	2	0	0,000	2	2	0,917	0	0	0,000	0,767	6	4
MTBLS267	2	2	0,618	2	0	0,000	2	2	0,917	0	0	0,000	0,767	6	4
MTBLS270	3	1	0,581	3	2	1,000	2	2	0,917	0	0	0,000	0,883	8	5
MTBLS276	3	2	0,828	1	1	1,000	2	2	0,917	0	0	0,000	0,898	6	5
MTBLS28	3	0	0,000	4	1	0,000	2	2	0,917	0	0	0,000	0,611	9	3
MTBLS281	4	4	0,846	2	2	0,500	2	2	0,917	0	0	0,000	0,777	8	8
MTBLS29	2	1	0,800	5	2	0,677	2	2	0,917	0	0	0,000	0,798	9	5
MTBLS293	3	2	0,693	2	1	1,000	2	2	0,917	0	0	0,000	0,844	7	5
MTBLS295	3	2	0,693	1	1	0,000	2	2	0,917	0	0	0,000	0,644	6	5
MTBLS3	4	3	0,827	1	1	1,000	2	2	0,950	0	0	0,000	0,897	7	6
MTBLS30	1	1	0,648	4	2	0,900	2	2	0,917	0	0	0,000	0,856	7	5
MTBLS31	4	1	0,242	3	3	0,868	2	2	0,917	0	0	0,000	0,780	9	6
MTBLS315	8	4	0,829	14	11	0,757	2	2	0,917	1	0	0,000	0,793	25	17
MTBLS32	4	1	0,242	3	3	0,868	2	2	0,917	0	0	0,000	0,780	9	6
MTBLS33	4	1	0,242	3	3	0,868	2	2	0,917	0	0	0,000	0,780	9	6
MTBLS34	4	1	0,242	3	3	0,868	2	2	0,917	0	0	0,000	0,780	9	6
MTBLS35	0	0	0,000	0	0	0,000	0	0	0,000	0	0	0,000	0,000	0	0
MTBLS36	0	0	0,000	1	1	1,000	2	2	0,950	0	0	0,000	0,967	3	3
MTBLS37	0	0	0,000	0	0	0,000	2	2	0,917	0	0	0,000	0,917	2	2
MTBLS38	0	0	0,000	0	0	0,000	0	0	0,000	0	0	0,000	0,000	0	0
MTBLS39	0	0	0,000	0	0	0,000	2	2	0,917	0	0	0,000	0,917	2	2
MTBLS4	4	2	1,000	1	0	0,000	2	2	0,917	0	0	0,000	0,958	7	4
MTBLS40	0	0	0,000	0	0	0,000	0	0	0,000	0	0	0,000	0,000	0	0
MTBLS41	4	2	0,414	3	1	0,702	2	2	0,917	1	0	0,000	0,673	10	5
MTBLS42	4	2	0,414	3	1	0,702	2	2	0,917	1	0	0,000	0,673	10	5
MTBLS43	4	2	0,414	0	0	0,000	2	2	0,917	1	0	0,000	0,665	7	4
MTBLS44	4	2	0,414	0	0	0,000	2	2	0,917	1	0	0,000	0,665	7	4
MTBLS45	0	0	0,000	0	0	0,000	2	2	0,917	0	0	0,000	0,917	2	2
MTBLS46	0	0	0,000	0	0	0,000	0	0	0,000	0	0	0,000	0,000	0	0
MTBLS47	0	0	0,000	0	0	0,000	2	2	0,917	0	0	0,000	0,917	2	2
MTBLS49	3	2	0,255	2	0	0,000	2	2	0,917	0	0	0,000	0,586	7	4
MTBLS5	3	1	0,648	1	1	0,000	2	2	0,917	1	0	0,000	0,620	7	4
MTBLS52	4	2	0,790	1	0	0,000	2	2	0,917	0	0	0,000	0,853	7	4
MTBLS54	4	2	0,414	0	0	0,000	2	2	0,917	1	0	0,000	0,665	7	4
MTBLS55	3	2	1,000	0	0	0,000	2	2	0,917	0	0	0,000	0,958	5	4
MTBLS56	0	0	0,000	0	0	0,000	2	2	1,000	0	0	0,000	1,000	2	2
MTBLS57	0	0	0,000	0	0	0,000	2	2	0,917	0	0	0,000	0,917	2	2
MTBLS59	0	0	0,000	0	0	0,000	0	0	0,000	0	0	0,000	0,000	0	0
MTBLS6	3	2	0,450	0	0	0,000	2	2	0,917	0	0	0,000	0,683	5	4
MTBLS61	0	0	0,000	0	0	0,000	0	0	0,000	0	0	0,000	0,000	0	0
MTBLS67	4	4	0,649	1	1	0,750	2	2	0,917	0	0	0,000	0,740	7	7
MTBLS69	0	0	0,000	0	0	0,000	2	2	0,917	0	0	0,000	0,917	2	2
MTBLS7	4	2	0,824	3	2	0,851	2	2	0,917	1	0	0,000	0,864	10	6
MTBLS71	4	1	0,000	2	1	1,000	2	2	0,917	0	0	0,000	0,708	8	4
MTBLS72	6	4	0,914	0	0	0,000	2	2	0,917	1	0	0,000	0,915	9	6
MTBLS74	3	3	0,917	1	1	1,000	2	2	0,950	0	0	0,000	0,942	6	6
MTBLS75	10	4	0,964	3	2	1,000	2	2	1,000	0	0	0,000	0,982	15	8
MTBLS77	6	4	0,917	3	2	0,800	2	2	1,000	0	0	0,000	0,908	11	8
MTBLS79	2	1	0,875	1	1	1,000	2	2	0,917	0	0	0,000	0,927	5	4
MTBLS8	2	2	0,324	2	0	0,000	2	2	0,917	1	0	0,000	0,620	7	4
MTBLS81	5	3	0,885	3	2	0,800	2	2	0,950	0	0	0,000	0,879	10	7
MTBLS85	3	2	0,938	1	0	0,000	2	2	0,950	0	0	0,000	0,944	6	4
MTBLS86	0	0	0,000	0	0	0,000	2	2	0,950	0	0	0,000	0,950	2	2
MTBLS87	2	2	0,618	1	1	0,714	2	2	0,917	0	0	0,000	0,757	5	5
MTBLS88	2	2	0,618	1	1	0,714	2	2	0,917	0	0	0,000	0,757	5	5
MTBLS90	3	1	1,000	5	2	0,814	2	2	0,917	1	0	0,000	0,892	11	5
MTBLS91	3	1	1,000	3	2	1,000	2	2	0,950	3	0	0,000	0,980	11	5
MTBLS92	3	1	0,000	1	1	1,000	2	2	0,950	0	0	0,000	0,725	6	4
MTBLS93	4	1	1,000	4	2	0,814	2	2	0,917	1	0	0,000	0,892	11	5
MTBLS95	7	3	0,868	2	1	1,000	2	2	0,917	0	0	0,000	0,906	11	6
MTBLS96	6	1	1,000	3	2	0,741	2	2	0,917	0	0	0,000	0,863	11	5
MTBLS99	1	1	0,511	3	0	0,000	2	2	0,917	0	0	0,000	0,781	6	3

Anexo K Resultados do MAA de metadados - Cobertura

Tabela 6-6 - Lista de resultados de cobertura por estudo do MAA.

	Design Class			Factor Class			Assay Class			Protocol Class			Overall		
Study ID	Found Anno.	Terms	Coverage	Found Anno.	Terms	Coverage	Found Anno.	Terms	Coverage	Found Anno.	Terms	Coverage	Avg. Coverage	Total Terms	Found Anno.
MTBLS1	3	6	0,500	1	2	0,500	2	2	1,000	0	10	0,000	0,300	20	6
MTBLS10	4	6	0,667	2	2	1,000	2	2	1,000	0	9	0,000	0,421	19	8
MTBLS100	1	2	0,500	0	1	0,000	2	2	1,000	0	11	0,000	0,188	16	3
MTBLS102	3	6	0,500	0	1	0,000	2	2	1,000	0	9	0,000	0,278	18	5
MTBLS103	0	1	0,000	0	1	0,000	3	3	1,000	0	14	0,000	0,158	19	3
MTBLS104	0	3	0,000	0	1	0,000	2	2	1,000	0	10	0,000	0,125	16	2
MTBLS105	2	6	0,333	1	2	0,500	2	2	1,000	0	10	0,000	0,250	20	5
MTBLS107	0	1	0,000	0	2	0,000	2	2	1,000	0	9	0,000	0,143	14	2
MTBLS108	0	1	0,000	0	2	0,000	2	2	1,000	0	9	0,000	0,143	14	2
MTBLS109	0	1	0,000	0	2	0,000	2	2	1,000	0	9	0,000	0,143	14	2
MTBLS11	2	5	0,400	2	4	0,500	2	2	1,000	0	9	0,000	0,300	20	6
MTBLS110	0	1	0,000	0	2	0,000	2	2	1,000	0	9	0,000	0,143	14	2
MTBLS111	0	1	0,000	0	2	0,000	2	2	1,000	0	9	0,000	0,143	14	2
MTBLS112	1	2	0,500	1	3	0,333	2	2	1,000	0	9	0,000	0,250	16	4
MTBLS113	2	5	0,400	3	6	0,500	2	2	1,000	0	10	0,000	0,304	23	7
MTBLS114	0	1	0,000	0	1	0,000	2	2	1,000	0	10	0,000	0,143	14	2
MTBLS116	5	6	0,833	0	4	0,000	2	2	1,000	0	11	0,000	0,304	23	7
MTBLS117	1	4	0,250	3	3	1,000	2	2	1,000	0	9	0,000	0,333	18	6
MTBLS118	1	3	0,333	3	3	1,000	2	2	1,000	0	9	0,000	0,353	17	6
MTBLS119	2	4	0,500	1	2	0,500	2	2	1,000	0	9	0,000	0,294	17	5
MTBLS12	2	5	0,400	2	4	0,500	2	2	1,000	0	9	0,000	0,300	20	6
MTBLS120	2	4	0,500	1	2	0,500	2	2	1,000	0	9	0,000	0,294	17	5
MTBLS123	2	3	0,667	3	3	1,000	2	2	1,000	0	10	0,000	0,389	18	7
MTBLS124	0	2	0,000	0	7	0,000	2	2	1,000	0	10	0,000	0,095	21	2
MTBLS125	3	6	0,500	1	2	0,500	2	2	1,000	0	9	0,000	0,316	19	6
MTBLS126	2	5	0,400	1	2	0,500	2	2	1,000	0	9	0,000	0,278	18	5
MTBLS127	2	5	0,400	2	2	1,000	2	2	1,000	0	9	0,000	0,333	18	6
MTBLS128	2	6	0,333	2	2	1,000	2	2	1,000	0	10	0,000	0,300	20	6
MTBLS13	2	5	0,400	2	4	0,500	2	2	1,000	0	9	0,000	0,300	20	6
MTBLS131	1	3	0,333	2	2	1,000	2	2	1,000	0	10	0,000	0,294	17	5
MTBLS132	1	3	0,333	2	2	1,000	2	2	1,000	0	10	0,000	0,294	17	5
MTBLS133	1	3	0,333	0	0	0,000	2	2	1,000	0	10	0,000	0,200	15	3
MTBLS134	1	0	0,000	0	0	0,000	2	0	0,000	0	0	0,000	0,000	0	3
MTBLS137	0	2	0,000	1	2	0,500	2	2	1,000	0	9	0,000	0,200	15	3
MTBLS14	2	5	0,400	3	7	0,429	2	2	1,000	0	9	0,000	0,304	23	7
MTBLS140	0	4	0,000	3	3	1,000	2	2	1,000	0	9	0,000	0,278	18	5
MTBLS143	1	10	0,100	2	2	1,000	2	2	1,000	0	9	0,000	0,217	23	5
MTBLS144	2	5	0,400	2	2	1,000	2	2	1,000	0	9	0,000	0,333	18	6
MTBLS146	3	0	0,000	2	0	0,000	2	0	0,000	0	0	0,000	0,000	0	7
MTBLS147	2	0	0,000	0	0	0,000	2	0	0,000	1	0	0,000	0,000	0	5
MTBLS148	2	3	0,667	6	7	0,857	2	2	1,000	0	9	0,000	0,476	21	10
MTBLS15	2	5	0,400	3	7	0,429	2	2	1,000	0	9	0,000	0,304	23	7
MTBLS150	1	5	0,200	2	3	0,667	2	2	1,000	0	9	0,000	0,263	19	5
MTBLS152	1	5	0,200	2	2	1,000	2	2	1,000	0	9	0,000	0,278	18	5
MTBLS154	2	6	0,333	3	3	1,000	2	2	1,000	0	9	0,000	0,350	20	7
MTBLS155	2	6	0,333	2	2	1,000	2	2	1,000	0	9	0,000	0,316	19	6
MTBLS156	0	4	0,000	0	0	0,000	2	2	1,000	0	10	0,000	0,125	16	2
MTBLS157	2	5	0,400	3	3	1,000	2	2	1,000	0	9	0,000	0,368	19	7
MTBLS16	2	5	0,400	3	7	0,429	2	2	1,000	0	9	0,000	0,304	23	7
MTBLS161	4	7	0,571	1	2	0,500	2	2	1,000	0	10	0,000	0,333	21	7
MTBLS162	1	5	0,200	3	4	0,750	2	2	1,000	0	9	0,000	0,300	20	6
MTBLS163	1	6	0,167	0	0	0,000	2	2	1,000	0	9	0,000	0,176	17	3
MTBLS165	1	3	0,333	2	3	0,667	2	2	1,000	0	9	0,000	0,294	17	5
MTBLS166	1	3	0,333	0	2	0,000	2	2	1,000	0	14	0,000	0,143	21	3
MTBLS168	0	0	0,000	1	2	0,500	2	2	1,000	0	9	0,000	0,231	13	3
MTBLS169	3	3	1,000	1	1	1,000	2	2	1,000	0	10	0,000	0,375	16	6
MTBLS17	3	4	0,750	3	3	1,000	2	2	1,000	0	9	0,000	0,444	18	8
MTBLS170	2	3	0,667	1	1	1,000	2	2	1,000	0	9	0,000	0,333	15	5
MTBLS171	3	4	0,750	1	2	0,500	2	2	1,000	0	9	0,000	0,353	17	6
MTBLS172	2	4	0,500	0	0	0,000	2	2	1,000	0	10	0,000	0,250	16	4
MTBLS173	4	6	0,667	3	3	1,000	2	2	1,000	0	9	0,000	0,450	20	9
MTBLS174	2	6	0,333	4	5	0,800	2	2	1,000	0	10	0,000	0,348	23	8
MTBLS175	1	4	0,250	0	2	0,000	2	2	1,000	0	10	0,000	0,167	18	3
MTBLS176	2	3	0,667	0	5	0,000	2	2	1,000	0	14	0,000	0,167	24	4
MTBLS177	3	7	0,429	2	3	0,667	2	2	1,000	0	9	0,000	0,333	21	7
MTBLS178	1	4	0,250	0	3	0,000	2	2	1,000	0	9	0,000	0,167	18	3
MTBLS187	2	6	0,333	3	3	1,000	2	2	1,000	0	9	0,000	0,350	20	7
MTBLS188	2	2	1,000	2	2	1,000	2	2	1,000	0	9	0,000	0,400	15	6
MTBLS19	3	4	0,750	3	3	1,000	2	2	1,000	0	9	0,000	0,444	18	8
MTBLS191	3	7	0,429	2	2	1,000	2	2	1,000	0	9	0,000	0,350	20	7
MTBLS194	0	3	0,000	0	0	0,000	2	2	1,000	0	9	0,000	0,143	14	2
MTBLS197	3	11	0,273	1	1	1,000	2	2	1,000	0	9	0,000	0,261	23	6
MTBLS2	1	5	0,200	2	2	1,000	2	2	1,000	0	9	0,000	0,278	18	5
MTBLS20	3	6	0,500	1	1	1,000	2	2	1,000	0	9	0,000	0,333	18	6
MTBLS202	3	3	1,000	1	1	1,000	2	2	1,000	0	10	0,000	0,375	16	6
MTBLS203	2	6	0,333	1	3	0,333	2	2	1,000	0	9	0,000	0,250	20	5
MTBLS208	2	4	0,500	1	1	1,000	2	2	1,000	0	9	0,000	0,313	16	5
MTBLS21	2	5	0,400	1	2	0,500	2	2	1,000	0	9	0,000	0,278	18	5
MTBLS210	2	5	0,400	1	1	1,000	2	2	1,000	0	9	0,000	0,294	17	5

	Design Class			Factor Class			Assay Class			Protocol Class			Overall		
Study ID	Found Anno.	Terms	Coverage	Found Anno.	Terms	Coverage	Found Anno.	Terms	Coverage	Found Anno.	Terms	Coverage	Avg. Coverage	Total Terms	Found Anno.
MTBLS212	4	8	0,500	1	2	0,500	2	2	1,000	0	9	0,000	0,333	21	7
MTBLS213	1	4	0,250	1	2	0,500	2	2	1,000	0	9	0,000	0,235	17	4
MTBLS214	1	3	0,333	2	2	1,000	2	2	1,000	0	9	0,000	0,313	16	5
MTBLS215	2	3	0,667	1	1	1,000	2	2	1,000	0	9	0,000	0,333	15	5
MTBLS218	2	4	0,500	1	2	0,500	2	2	1,000	0	9	0,000	0,294	17	5
MTBLS219	2	3	0,667	1	2	0,500	2	2	1,000	0	9	0,000	0,313	16	5
MTBLS22	2	5	0,400	1	4	0,250	2	2	1,000	0	9	0,000	0,250	20	5
MTBLS226	3	5	0,600	2	3	0,667	2	2	1,000	0	9	0,000	0,368	19	7
MTBLS228	0	4	0,000	1	1	1,000	2	2	1,000	0	9	0,000	0,188	16	3
MTBLS229	0	4	0,000	1	1	1,000	2	2	1,000	0	9	0,000	0,188	16	3
MTBLS23	1	6	0,167	0	1	0,000	2	2	1,000	0	9	0,000	0,167	18	3
MTBLS234	2	3	0,667	2	2	1,000	2	2	1,000	0	9	0,000	0,375	16	6
MTBLS235	1	3	0,333	0	3	0,000	2	2	1,000	0	9	0,000	0,176	17	3
MTBLS24	1	3	0,333	2	2	1,000	2	2	1,000	0	10	0,000	0,294	17	5
MTBLS243	5	6	0,833	0	1	0,000	2	2	1,000	0	9	0,000	0,389	18	7
MTBLS25	1	4	0,250	0	0	0,000	2	2	1,000	0	12	0,000	0,167	18	3
MTBLS26	3	9	0,333	1	2	0,500	2	2	1,000	0	10	0,000	0,261	23	6
MTBLS263	2	5	0,400	2	2	1,000	2	2	1,000	0	9	0,000	0,333	18	6
MTBLS264	2	5	0,400	1	3	0,333	2	2	1,000	0	9	0,000	0,263	19	5
MTBLS265	2	4	0,500	0	2	0,000	2	2	1,000	0	9	0,000	0,235	17	4
MTBLS266	2	4	0,500	0	2	0,000	2	2	1,000	0	9	0,000	0,235	17	4
MTBLS267	2	4	0,500	0	2	0,000	2	2	1,000	0	9	0,000	0,235	17	4
MTBLS270	1	4	0,250	2	3	0,667	2	2	1,000	0	9	0,000	0,278	18	5
MTBLS276	2	4	0,500	1	2	0,500	2	2	1,000	0	9	0,000	0,294	17	5
MTBLS28	0	5	0,000	1	4	0,250	2	2	1,000	0	9	0,000	0,150	20	3
MTBLS281	4	7	0,571	2	2	1,000	2	2	1,000	0	9	0,000	0,400	20	8
MTBLS29	1	0	0,000	2	0	0,000	2	0	0,000	0	0	0,000	0,000	0	5
MTBLS293	2	5	0,400	1	2	0,500	2	2	1,000	0	9	0,000	0,278	18	5
MTBLS295	2	5	0,400	1	1	1,000	2	2	1,000	0	9	0,000	0,294	17	5
MTBLS3	3	6	0,500	1	1	1,000	2	2	1,000	0	9	0,000	0,333	18	6
MTBLS30	1	6	0,167	2	4	0,500	2	2	1,000	0	9	0,000	0,238	21	5
MTBLS31	1	5	0,200	3	3	1,000	2	2	1,000	0	9	0,000	0,316	19	6
MTBLS315	4	10	0,400	11	20	0,550	2	2	1,000	0	9	0,000	0,415	41	17
MTBLS32	1	5	0,200	3	3	1,000	2	2	1,000	0	9	0,000	0,316	19	6
MTBLS33	1	5	0,200	3	3	1,000	2	2	1,000	0	9	0,000	0,316	19	6
MTBLS34	1	5	0,200	3	3	1,000	2	2	1,000	0	9	0,000	0,316	19	6
MTBLS35	0	6	0,000	0	3	0,000	0	0	0,000	0	10	0,000	0,000	19	0
MTBLS36	0	2	0,000	1	3	0,333	2	2	1,000	0	10	0,000	0,176	17	3
MTBLS37	0	1	0,000	0	2	0,000	2	2	1,000	0	9	0,000	0,143	14	2
MTBLS38	0	2	0,000	0	0	0,000	0	2	0,000	0	9	0,000	0,000	13	0
MTBLS39	0	3	0,000	0	2	0,000	2	2	1,000	0	9	0,000	0,125	16	2
MTBLS4	2	6	0,333	0	1	0,000	2	2	1,000	0	9	0,000	0,222	18	4
MTBLS40	0	4	0,000	0	1	0,000	0	2	0,000	0	10	0,000	0,000	17	0
MTBLS41	2	5	0,400	1	4	0,250	2	2	1,000	0	9	0,000	0,250	20	5
MTBLS42	2	5	0,400	1	4	0,250	2	2	1,000	0	9	0,000	0,250	20	5
MTBLS43	2	5	0,400	0	1	0,000	2	2	1,000	0	9	0,000	0,235	17	4
MTBLS44	2	5	0,400	0	1	0,000	2	2	1,000	0	9	0,000	0,235	17	4
MTBLS45	0	5	0,000	0	1	0,000	2	2	1,000	0	11	0,000	0,105	19	2
MTBLS46	0	3	0,000	0	2	0,000	0	0	0,000	0	8	0,000	0,000	13	0
MTBLS47	0	4	0,000	0	2	0,000	2	2	1,000	0	10	0,000	0,111	18	2
MTBLS49	2	3	0,667	0	2	0,000	2	2	1,000	0	9	0,000	0,250	16	4
MTBLS5	1	4	0,250	1	1	1,000	2	2	1,000	0	9	0,000	0,250	16	4
MTBLS52	2	4	0,500	0	3	0,000	2	2	1,000	0	9	0,000	0,222	18	4
MTBLS54	2	5	0,400	0	1	0,000	2	2	1,000	0	9	0,000	0,235	17	4
MTBLS55	2	7	0,286	0	3	0,000	2	2	1,000	0	10	0,000	0,182	22	4
MTBLS56	0	5	0,000	0	2	0,000	2	2	1,000	0	10	0,000	0,105	19	2
MTBLS57	0	5	0,000	0	1	0,000	2	2	1,000	0	9	0,000	0,118	17	2
MTBLS59	0	3	0,000	0	3	0,000	0	2	0,000	0	8	0,000	0,000	16	0
MTBLS6	2	4	0,500	0	9	0,000	2	2	1,000	0	9	0,000	0,167	24	4
MTBLS61	0	5	0,000	0	3	0,000	0	2	0,000	0	9	0,000	0,000	19	0
MTBLS67	4	5	0,800	1	1	1,000	2	2	1,000	0	9	0,000	0,412	17	7
MTBLS69	0	2	0,000	0	1	0,000	2	2	1,000	0	9	0,000	0,143	14	2
MTBLS7	2	5	0,400	2	4	0,500	2	2	1,000	0	9	0,000	0,300	20	6
MTBLS71	1	4	0,250	1	2	0,500	2	2	1,000	0	12	0,000	0,200	20	4
MTBLS72	4	9	0,444	0	0	0,000	2	0	0,000	0	0	0,000	0,667	9	6
MTBLS74	3	3	1,000	1	1	1,000	2	2	1,000	0	10	0,000	0,375	16	6
MTBLS75	4	11	0,364	2	3	0,667	2	2	1,000	0	9	0,000	0,320	25	8
MTBLS77	4	7	0,571	2	3	0,667	2	2	1,000	0	10	0,000	0,364	22	8
MTBLS79	1	8	0,125	1	1	1,000	2	2	1,000	0	20	0,000	0,129	31	4
MTBLS8	2	5	0,400	0	2	0,000	2	2	1,000	0	9	0,000	0,222	18	4
MTBLS81	3	8	0,375	2	5	0,400	2	2	1,000	0	9	0,000	0,292	24	7
MTBLS85	2	4	0,500	0	2	0,000	2	2	1,000	0	10	0,000	0,222	18	4
MTBLS86	0	2	0,000	0	2	0,000	2	2	1,000	0	10	0,000	0,125	16	2
MTBLS87	2	4	0,500	1	1	1,000	2	2	1,000	0	9	0,000	0,313	16	5
MTBLS88	2	4	0,500	1	1	1,000	2	2	1,000	0	9	0,000	0,313	16	5
MTBLS90	1	3	0,333	2	7	0,286	2	2	1,000	0	10	0,000	0,227	22	5
MTBLS91	1	4	0,250	2	3	0,667	2	2	1,000	0	11	0,000	0,250	20	5
MTBLS92	1	4	0,250	1	10	0,100	2	2	1,000	0	9	0,000	0,160	25	4
MTBLS93	1	4	0,250	2	7	0,286	2	2	1,000	0	10	0,000	0,217	23	5
MTBLS95	3	7	0,429	1	2	0,500	2	2	1,000	0	9	0,000	0,300	20	6
MTBLS96	1	6	0,167	2	3	0,667	2	2	1,000	0	9	0,000	0,250	20	5
MTBLS99	1	1	1,000	0	5	0,000	2	2	1,000	0	9	0,000	0,176	17	3

Anexo L Resultados do MAA de metadados – Tempos de execução

Tabela 6-7 - Tempos de execução (em milissegundos) por estudos do Metabolights.

ID	Tot. Anno.	Tot. Terms	Avg. Specificity	Avg. Coverage	Run Time
MTBLS1	6	20	0,888	0,300	46084
MTBLS10	8	19	0,797	0,421	33685
MTBLS100	3	16	0,667	0,188	7137
MTBLS102	5	18	0,874	0,278	40815
MTBLS103	3	19	0,944	0,158	4301
MTBLS104	2	16	1,000	0,125	3857
MTBLS105	5	20	0,764	0,250	123681
MTBLS107	2	14	0,917	0,143	3913
MTBLS108	2	14	0,917	0,143	3918
MTBLS109	2	14	0,917	0,143	4569
MTBLS11	6	20	0,864	0,300	20972
MTBLS110	2	14	0,917	0,143	3392
MTBLS111	2	14	0,917	0,143	3991
MTBLS112	4	16	0,896	0,250	10890
MTBLS113	7	23	0,976	0,304	22413
MTBLS114	2	14	1,000	0,143	3926
MTBLS116	7	23	0,940	0,304	23098
MTBLS117	6	18	0,814	0,333	9254
MTBLS118	6	17	0,812	0,353	3712
MTBLS119	5	17	0,902	0,294	10201
MTBLS12	6	20	0,864	0,300	5545
MTBLS120	5	17	0,902	0,294	3716
MTBLS123	7	18	0,848	0,389	18039
MTBLS124	2	21	0,917	0,095	22990
MTBLS125	6	19	0,761	0,316	16670
MTBLS126	5	18	0,814	0,278	10151
MTBLS127	6	18	0,672	0,333	4272
MTBLS128	6	20	0,840	0,300	14108
MTBLS13	6	20	0,864	0,300	11913
MTBLS131	5	17	0,933	0,294	11100
MTBLS132	5	17	0,933	0,294	3906
MTBLS133	3	15	1,000	0,200	4777
MTBLS134	3	0	1,000	0,000	4249
MTBLS137	3	15	0,944	0,200	11566
MTBLS14	7	23	0,766	0,304	28044
MTBLS140	5	18	0,917	0,278	6271
MTBLS143	5	23	0,888	0,217	26028
MTBLS144	6	18	0,828	0,333	9569
MTBLS146	7	0	0,763	0,000	24527
MTBLS147	5	0	0,564	0,000	18144
MTBLS148	10	21	0,674	0,476	8748
MTBLS15	7	23	0,766	0,304	12273
MTBLS150	5	19	0,769	0,263	27095
MTBLS152	5	18	0,769	0,278	3983
MTBLS154	7	20	0,825	0,350	4800
MTBLS155	6	19	0,828	0,316	3929
MTBLS156	2	16	1,000	0,125	4593
MTBLS157	7	19	0,853	0,368	11732
MTBLS16	7	23	0,766	0,304	5321
MTBLS161	7	21	0,797	0,333	25522
MTBLS162	6	20	0,830	0,300	15120
MTBLS163	3	17	0,844	0,176	5924
MTBLS165	5	17	0,793	0,294	7343
MTBLS166	3	21	1,000	0,143	3829
MTBLS168	3	13	0,825	0,231	3894
MTBLS169	6	16	0,908	0,375	6668
MTBLS17	8	18	0,852	0,444	9585
MTBLS170	5	15	0,798	0,333	5389
MTBLS171	6	17	0,705	0,353	16441
MTBLS172	4	16	0,840	0,250	12308
MTBLS173	9	20	0,834	0,450	16846
MTBLS174	8	23	0,886	0,348	9750
MTBLS175	3	18	0,827	0,167	11582
MTBLS176	4	24	0,958	0,167	21425
MTBLS177	7	21	0,906	0,333	29306
MTBLS178	3	18	0,944	0,167	9644
MTBLS187	7	20	0,906	0,350	7649
MTBLS188	6	15	0,873	0,400	5110
MTBLS19	8	18	0,852	0,444	3301
MTBLS191	7	20	0,783	0,350	9672
MTBLS194	2	14	0,917	0,143	3369
MTBLS197	6	23	0,666	0,261	37536
MTBLS2	5	18	0,940	0,278	3823
MTBLS20	6	18	0,867	0,333	7146
MTBLS202	6	16	0,983	0,375	11717
MTBLS203	5	20	0,967	0,250	4469
MTBLS208	5	16	0,846	0,313	3726
MTBLS21	5	18	0,858	0,278	4846
MTBLS210	5	17	0,858	0,294	7851
MTBLS212	7	21	0,912	0,333	25537

ID	Tot. Anno.	Tot. Terms	Avg. Specificity	Avg. Coverage	Run Time
MTBLS213	4	17	0,910	0,235	18517
MTBLS214	5	16	0,791	0,313	16852
MTBLS215	5	15	0,630	0,333	4916
MTBLS218	5	17	0,967	0,294	12651
MTBLS219	5	16	0,863	0,313	8963
MTBLS22	5	20	0,673	0,250	6326
MTBLS226	7	19	0,816	0,368	13302
MTBLS228	3	16	0,944	0,188	3510
MTBLS229	3	16	0,944	0,188	4258
MTBLS23	3	18	0,827	0,167	4074
MTBLS234	6	16	0,940	0,375	4950
MTBLS235	3	17	0,944	0,176	3385
MTBLS24	5	17	0,866	0,294	6639
MTBLS243	7	18	0,692	0,389	21536
MTBLS25	3	18	0,860	0,167	8721
MTBLS26	6	23	0,893	0,261	13171
MTBLS263	6	18	0,845	0,333	14519
MTBLS264	5	19	0,764	0,263	16334
MTBLS265	4	17	0,767	0,235	8839
MTBLS266	4	17	0,767	0,235	3932
MTBLS267	4	17	0,767	0,235	4128
MTBLS270	5	18	0,883	0,278	8821
MTBLS276	5	17	0,898	0,294	11434
MTBLS28	3	20	0,611	0,150	13312
MTBLS281	8	20	0,777	0,400	5455
MTBLS29	5	0	0,798	0,000	14695
MTBLS293	5	18	0,844	0,278	3740
MTBLS295	5	17	0,644	0,294	4204
MTBLS3	6	18	0,897	0,333	4125
MTBLS30	5	21	0,856	0,238	12681
MTBLS31	6	19	0,780	0,316	27237
MTBLS315	17	41	0,793	0,415	38809
MTBLS32	6	19	0,780	0,316	3684
MTBLS33	6	19	0,780	0,316	4207
MTBLS34	6	19	0,780	0,316	14018
MTBLS35	0	19	0,000	0,000	3416
MTBLS36	3	17	0,967	0,176	4302
MTBLS37	2	14	0,917	0,143	4492
MTBLS38	0	13	0,000	0,000	4421
MTBLS39	2	16	0,917	0,125	3909
MTBLS4	4	18	0,958	0,222	4908
MTBLS40	0	17	0,000	0,000	3979
MTBLS41	5	20	0,673	0,250	4358
MTBLS42	5	20	0,673	0,250	9595
MTBLS43	4	17	0,665	0,235	7777
MTBLS44	4	17	0,665	0,235	3530
MTBLS45	2	19	0,917	0,105	3954
MTBLS46	0	13	0,000	0,000	4516
MTBLS47	2	18	0,917	0,111	3548
MTBLS49	4	16	0,586	0,250	17440
MTBLS5	4	16	0,620	0,250	10883
MTBLS52	4	18	0,853	0,222	4473
MTBLS54	4	17	0,665	0,235	3775
MTBLS55	4	22	0,958	0,182	5122
MTBLS56	2	19	1,000	0,105	3625
MTBLS57	2	17	0,917	0,118	4142
MTBLS59	0	16	0,000	0,000	4322
MTBLS6	4	24	0,683	0,167	8582
MTBLS61	0	19	0,000	0,000	3413
MTBLS67	7	17	0,740	0,412	12482
MTBLS69	2	14	0,917	0,143	3747
MTBLS7	6	20	0,864	0,300	4240
MTBLS71	4	20	0,708	0,200	9591
MTBLS72	6	9	0,915	0,667	21280
MTBLS74	6	16	0,942	0,375	9129
MTBLS75	8	25	0,982	0,320	17697
MTBLS77	8	22	0,908	0,364	15882
MTBLS79	4	31	0,927	0,129	3605
MTBLS8	4	18	0,620	0,222	7204
MTBLS81	7	24	0,879	0,292	3770
MTBLS85	4	18	0,944	0,222	16427
MTBLS86	2	16	0,950	0,125	3509
MTBLS87	5	16	0,757	0,313	4100
MTBLS88	5	16	0,757	0,313	4264
MTBLS90	5	22	0,892	0,227	24047
MTBLS91	5	20	0,980	0,250	5437
MTBLS92	4	25	0,725	0,160	7712
MTBLS93	5	23	0,892	0,217	8772
MTBLS95	6	20	0,906	0,300	29509
MTBLS96	5	20	0,863	0,250	6193
MTBLS99	3	17	0,781	0,176	15540

Anexo M Resultados do MAA de metadados – Lista de termos encontrados

Tabela 6-8 - Lista de anotações do repositório Metabolights.

Annotation	Used Count	Meta Spec	Proc Spec	Run Time
http://purl.obolibrary.org/obo/OBI_0000366	161	1,000	1,000	1199
http://purl.obolibrary.org/obo/OBI_0000623	27	1,000	1,000	1419
http://purl.obolibrary.org/obo/CHMO_0000715	25	1,000	1,000	1503
http://www.ebi.ac.uk/efo/EFO_0000752	13	1,000	1,000	7401
http://www.ebi.ac.uk/efo/EFO_0000433	12	1,000	1,000	4799
http://www.ebi.ac.uk/efo/EFO_0002091	9	1,000	1,000	5103
http://www.ebi.ac.uk/efo/EFO_0002090	8	1,000	1,000	4800
http://www.ebi.ac.uk/efo/EFO_0001702	7	1,000	1,000	5800
http://www.ebi.ac.uk/efo/EFO_0000428	7	1,000	1,000	5896
http://www.ebi.ac.uk/efo/EFO_0000689	6	1,000	1,000	7782
http://www.ebi.ac.uk/efo/EFO_0000544	5	1,000	1,000	4799
http://purl.obolibrary.org/obo/CHMO_0002443	4	1,000	1,000	1300
http://www.ebi.ac.uk/efo/EFO_0001461	4	1,000	1,000	5000
http://www.ebi.ac.uk/efo/EFO_0000195	4	1,000	1,000	5999
http://purl.obolibrary.org/obo/BTO_0004531	4	1,000	1,000	1700
http://www.ebi.ac.uk/efo/EFO_0001776	3	1,000	1,000	5806
http://www.ebi.ac.uk/efo/EFO_0001422	3	1,000	1,000	4599
http://www.ebi.ac.uk/efo/EFO_0005067	3	1,000	1,000	1598
http://www.ebi.ac.uk/efo/EFO_0002756	2	1,000	1,000	3899
http://purl.obolibrary.org/obo/UO_0000032	2	1,000	1,000	1088
http://purl.obolibrary.org/obo/UO_0000033	2	1,000	1,000	1516
http://purl.obolibrary.org/obo/OBI_0200199	2	1,000	1,000	1495
http://purl.obolibrary.org/obo/MS_1000058	2	1,000	1,000	1200
http://purl.obolibrary.org/obo/MS_1001808	2	1,000	1,000	894
http://www.ebi.ac.uk/efo/EFO_0001715	2	1,000	1,000	4801
http://www.ebi.ac.uk/efo/EFO_0000506	2	1,000	1,000	5698
http://purl.obolibrary.org/obo/CHMO_0002886	2	1,000	1,000	995
http://www.ebi.ac.uk/efo/EFO_0000721	2	1,000	1,000	2300
http://purl.obolibrary.org/obo/MS_1001834	2	1,000	1,000	898
http://purl.obolibrary.org/obo/ENVO_00000150	2	1,000	1,000	2998
http://purl.obolibrary.org/obo/GO_0030257	1	1,000	1,000	9499
http://www.bioassayontology.org/bao#BAO_0000453	1	1,000	1,000	2701
http://purl.obolibrary.org/obo/OBI_0001627	1	1,000	1,000	4401
http://www.bioassayontology.org/bao#BAO_0002084	1	1,000	1,000	1200
http://purl.obolibrary.org/obo/MS_1001582	1	1,000	1,000	1201
http://www.bioassayontology.org/bao#BAO_0003063	1	1,000	1,000	1484
http://www.ebi.ac.uk/efo/EFO_0001779	1	1,000	1,000	3495
http://purl.obolibrary.org/obo/XCO_0000072	1	1,000	1,000	4004
http://purl.bioontology.org/ontology/HL7/C1550601	1	1,000	1,000	10601
http://purl.obolibrary.org/obo/BTO_0000133	1	1,000	1,000	2100
http://www.ebi.ac.uk/efo/EFO_0002757	1	1,000	1,000	7200
http://purl.obolibrary.org/obo/OBI_0001546	1	1,000	1,000	1999
http://www.ebi.ac.uk/efo/EFO_0001420	1	1,000	1,000	3700
http://purl.obolibrary.org/obo/OBI_0200196	1	1,000	1,000	2096
http://purl.obolibrary.org/obo/UO_0000031	1	1,000	1,000	3612
http://purl.obolibrary.org/obo/PATO_0001657	1	1,000	1,000	2307
http://purl.obolibrary.org/obo/UO_0000027	1	1,000	1,000	1261
http://www.bioassayontology.org/bao#BAO_0002441	1	1,000	1,000	2086
http://purl.obolibrary.org/obo/CHMO_0001585	1	1,000	1,000	1502
http://purl.obolibrary.org/obo/UO_0000169	1	1,000	1,000	999
http://www.bioassayontology.org/bao#BAO_0190003	1	1,000	1,000	2095
http://purl.obolibrary.org/obo/MS_1000045	1	1,000	1,000	1603
http://www.ebi.ac.uk/efo/EFO_0000249	1	1,000	1,000	6097
http://www.ebi.ac.uk/efo/EFO_0005135	1	1,000	1,000	2905
http://www.ebi.ac.uk/efo/EFO_0000483	1	1,000	1,000	1753
http://purl.obolibrary.org/obo/CHMO_0000593	1	1,000	1,000	1103
http://purl.obolibrary.org/obo/CMO_0000105	1	1,000	1,000	5341
http://purl.obolibrary.org/obo/CHMO_0000484	1	1,000	1,000	1058
http://purl.obolibrary.org/obo/CHMO_0002542	1	1,000	1,000	1100
http://purl.obolibrary.org/obo/CHMO_0002758	1	1,000	1,000	1398
http://www.ebi.ac.uk/efo/EFO_0000638	1	1,000	1,000	2802
http://www.ebi.ac.uk/efo/EFO_0000720	1	1,000	1,000	7401
http://purl.obolibrary.org/obo/CHMO_0000795	1	1,000	1,000	994
http://purl.obolibrary.org/obo/CHMO_0002852	1	1,000	1,000	1099
http://purl.obolibrary.org/obo/OBI_0001140	1	1,000	1,000	1301
http://www.ebi.ac.uk/efo/EFO_0001265	1	1,000	1,000	2997
http://www.ebi.ac.uk/efo/EFO_0001266	1	1,000	1,000	2899
http://www.bioassayontology.org/bao#BAO_0000055	1	1,000	1,000	1001
http://www.ebi.ac.uk/efo/EFO_0001949	1	1,000	1,000	2898
http://purl.obolibrary.org/obo/CHMO_0002503	1	1,000	1,000	1101
http://purl.obolibrary.org/obo/ENVO_01000161	1	1,000	1,000	6801
http://www.ebi.ac.uk/efo/EFO_0000546	1	1,000	1,000	6095
http://purl.obolibrary.org/obo/OBI_0001310	1	1,000	1,000	1299
http://purl.obolibrary.org/obo/CHMO_0001624	1	1,000	1,000	999
http://www.ebi.ac.uk/efo/EFO_0000676	1	1,000	1,000	3839
http://purl.bioontology.org/ontology/HL7/C0033086	1	1,000	1,000	4502
http://www.ebi.ac.uk/efo/EFO_0001760	1	1,000	1,000	4801
http://www.ebi.ac.uk/efo/EFO_0002950	1	1,000	1,000	4903

Annotation	Used Count	Meta Spec	Proc Spec	Run Time
http://www.ebi.ac.uk/efo/EFO_0003809	1	1,000	1,000	3400
http://purl.obolibrary.org/obo/GO_0045208	1	0,955	0,955	10668
http://purl.obolibrary.org/obo/OBI_0001478	13	0,900	0,900	1801
http://purl.obolibrary.org/obo/OBI_0000952	1	0,889	0,889	1494
http://purl.obolibrary.org/obo/CHMO_0000502	9	0,875	0,875	1690
http://purl.obolibrary.org/obo/CHMO_0002262	2	0,875	0,875	1900
http://purl.obolibrary.org/obo/CHMO_0001009	1	0,875	0,875	1801
http://www.ebi.ac.uk/efo/EFO_0003934	1	0,875	0,875	3400
http://purl.obolibrary.org/obo/ENVO_01000008	2	0,857	0,857	4202
http://purl.obolibrary.org/obo/BTO_0000093	1	0,857	0,857	1899
http://purl.obolibrary.org/obo/OBI_0000470	120	0,833	0,833	1096
http://www.ebi.ac.uk/efo/EFO_0001073	3	0,833	0,833	5001
http://www.ebi.ac.uk/efo/EFO_0004340	2	0,833	0,833	6301
http://www.ebi.ac.uk/efo/EFO_0002755	1	0,833	0,833	7187
http://www.ebi.ac.uk/efo/EFO_0004339	1	0,833	0,833	4361
http://purl.obolibrary.org/obo/CHMO_0002820	7	0,807	0,807	1600
http://www.ebi.ac.uk/efo/EFO_0000513	10	0,800	0,800	4499
http://www.ebi.ac.uk/efo/EFO_0000246	7	0,800	0,800	5000
http://purl.obolibrary.org/obo/CHMO_0001586	1	0,800	0,800	1512
http://purl.obolibrary.org/obo/CHMO_0001677	1	0,800	0,800	1094
http://purl.obolibrary.org/obo/CHMO_0000520	1	0,800	0,800	1401
http://www.ebi.ac.uk/efo/EFO_0005842	1	0,778	0,778	6497
http://www.ebi.ac.uk/efo/EFO_0005856	1	0,774	0,774	4004
http://purl.bioontology.org/NEMO/ontology/NEMO.owl#NEMO_5159000	1	0,764	0,764	3362
http://www.ebi.ac.uk/efo/EFO_0000182	3	0,759	0,759	10260
http://www.ebi.ac.uk/efo/EFO_0000724	20	0,750	0,750	2497
http://purl.obolibrary.org/obo/CHMO_0000580	3	0,750	0,750	1103
http://www.ebi.ac.uk/efo/EFO_0000400	2	0,750	0,750	2697
http://purl.obolibrary.org/obo/OBI_0000272	1	0,750	0,750	1197
http://purl.obolibrary.org/obo/OBI_0000747	3	0,714	0,714	2932
http://www.ebi.ac.uk/efo/EFO_0004338	1	0,714	0,714	4000
http://www.ebi.ac.uk/efo/EFO_0000579	10	0,702	0,702	6000
http://purl.obolibrary.org/obo/CHMO_0000506	1	0,700	0,700	4297
http://purl.obolibrary.org/obo/CHMO_0000701	1	0,700	0,700	1599
http://purl.obolibrary.org/obo/PO_0025500	2	0,667	0,667	2487
http://www.ebi.ac.uk/efo/EFO_0000355	1	0,667	0,667	7000
http://purl.obolibrary.org/obo/CHMO_0000491	1	0,667	0,667	1411
http://www.ebi.ac.uk/efo/EFO_0002460	1	0,667	0,667	4661
http://www.ebi.ac.uk/efo/EFO_0001068	1	0,667	0,667	6499
http://purl.obolibrary.org/obo/CHMO_0000575	18	0,656	0,656	1299
http://purl.obolibrary.org/obo/CHMO_0000497	31	0,648	0,648	1987
http://www.ebi.ac.uk/efo/EFO_0000683	5	0,643	0,643	9599
http://purl.obolibrary.org/obo/MS_1000294	2	0,629	0,629	2288
http://www.ebi.ac.uk/efo/EFO_0000727	7	0,605	0,605	5704
http://purl.obolibrary.org/obo/CHMO_0002743	1	0,600	0,600	1356
http://purl.obolibrary.org/obo/CHMO_0000796	17	0,581	0,581	1101
http://purl.obolibrary.org/obo/CHMO_0000591	8	0,580	0,580	3701
http://purl.obolibrary.org/obo/BTO_0000214	1	0,519	0,519	2299
http://purl.obolibrary.org/obo/GO_0015893	1	0,517	0,517	9798
http://purl.obolibrary.org/obo/CHMO_0000524	11	0,511	0,511	1899
http://purl.obolibrary.org/obo/UO_0000205	1	0,500	0,500	802
http://www.ebi.ac.uk/efo/EFO_0000311	2	0,482	0,482	16765
http://purl.obolibrary.org/obo/UO_0000003	1	0,482	0,482	898
http://purl.obolibrary.org/obo/PATO_0000125	1	0,450	0,450	2705
http://purl.obolibrary.org/obo/GO_0007568	1	0,407	0,407	17804
http://www.ebi.ac.uk/efo/EFO_0000322	3	0,368	0,368	21894
http://purl.obolibrary.org/obo/GO_0006995	1	0,359	0,359	7801
http://www.bioassayontology.org/bao#BAO_0000015	1	0,326	0,326	3903
http://purl.obolibrary.org/obo/GO_0006836	1	0,266	0,266	11202
http://www.ebi.ac.uk/efo/EFO_0000324	2	0,261	0,261	13098
http://purl.obolibrary.org/obo/MS_1001457	1	0,252	0,252	1910
http://purl.obolibrary.org/obo/GO_0006954	4	0,242	0,242	12303
http://www.ebi.ac.uk/efo/EFO_0000319	1	0,240	0,240	9900
http://www.ebi.ac.uk/efo/EFO_0000408	4	0,228	0,228	115695
http://purl.obolibrary.org/obo/GO_0009611	1	0,214	0,214	13599
http://purl.obolibrary.org/obo/GO_0009651	9	0,180	0,180	12298
http://purl.obolibrary.org/obo/GO_0042538	1	0,121	0,121	13500
http://purl.obolibrary.org/obo/MS_1000457	4	0,000	0,000	798
http://purl.obolibrary.org/obo/BTO_0000131	3	0,000	0,000	1798
http://purl.obolibrary.org/obo/BTO_0001419	3	0,000	0,000	1100
http://purl.obolibrary.org/obo/BTO_0001202	1	0,000	0,000	2308
http://purl.obolibrary.org/obo/MP_0001845	1	0,000	0,000	10301
http://purl.obolibrary.org/obo/MS_1000008	1	0,000	0,000	1303
http://purl.obolibrary.org/obo/MS_1000465	1	0,000	0,000	1799
http://purl.obolibrary.org/obo/BTO_0000316	1	0,000	0,000	1700
http://purl.obolibrary.org/obo/XCO_0000012	1	0,000	0,000	799
http://purl.obolibrary.org/obo/BTO_0001239	1	0,000	0,000	1600
http://purl.obolibrary.org/obo/SBO_0000000	1	0,000	0,000	2135
http://purl.bioontology.org/ontology/OBI/OBI_0000115	14	-1,000	-1,000	891

Annotation	Used Count	Meta Spec	Proc Spec	Run Time
http://purl.bioontology.org/ontology/OBI/OBI_0001396	14	-1,000	-1,000	1500
http://scai.fraunhofer.de/CSEO#lipidomics	12	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C17357	10	-1,000	-1,000	1253
http://purl.obolibrary.org/obo/XEO_00040	7	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/XEO_00124	6	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D002338	5	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D010937	5	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D057486	4	-1,000	-1,000	-1
http://www.owl-ontologies.com/unnamed.owl#Phytoplankton	4	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/NCBITaxon_11320	4	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C70668	4	-1,000	-1,000	4300
http://purl.bioontology.org/ontology/XEO/XEO:00040	3	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/SNOMEDCT/250829009	3	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C62709	3	-1,000	-1,000	2998
http://purl.bioontology.org/ontology/MESH/D015999	3	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C1215	3	-1,000	-1,000	4099
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C25164	3	-1,000	-1,000	2399
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C13195	3	-1,000	-1,000	3595
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C25150	3	-1,000	-1,000	3300
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C20587	3	-1,000	-1,000	3900
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=ba487064-8b46-4408-812c-b4e5083ad7c2	3	-1,000	-1,000	4799
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C15208	3	-1,000	-1,000	7300
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C68706	2	-1,000	-1,000	5399
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C12801	2	-1,000	-1,000	7900
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C62007	2	-1,000	-1,000	5800
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C38147	2	-1,000	-1,000	5700
http://purl.bioontology.org/ontology/MESH/D066292	2	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/OMIM_167000	2	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/NCBITaxon_7130	2	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C49019	2	-1,000	-1,000	4201
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=10b38aaf-b977-4950-85b8-f4775f66658d	2	-1,000	-1,000	3499
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C70665	2	-1,000	-1,000	7542
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=862e2c71-40f3-4bc0-9713-4432c5062cb8	2	-1,000	-1,000	4501
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C829	2	-1,000	-1,000	3401
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C25713	2	-1,000	-1,000	3201
http://purl.bioontology.org/ontology/MESH/C089796	2	-1,000	-1,000	-1
http://bioontology.org/ontologies/ResearchArea.owl#Metabolomics	2	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C17993	2	-1,000	-1,000	2107
http://purl.obolibrary.org/obo/FIX_0000359	2	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_18059	2	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C47933	2	-1,000	-1,000	7999
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#Population_Based_Study	2	-1,000	-1,000	4800
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C19796	1	-1,000	-1,000	6309
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=57545514-4007-4234-9ba0-f37fb2794081	1	-1,000	-1,000	3197
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C38872	1	-1,000	-1,000	4798
http://purl.bioontology.org/ontology/MESH/C007262	1	-1,000	-1,000	-1
http://www.mygrid.org.uk/ontology/JERMOntology#Metabonomics	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MSH/D008401	1	-1,000	-1,000	2192
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C28038	1	-1,000	-1,000	6789
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C22730	1	-1,000	-1,000	6712
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C25688	1	-1,000	-1,000	3488
http://genomics.uni-regensburg.de/site/institute/software/metaboquant	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C120538	1	-1,000	-1,000	8295
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C45293	1	-1,000	-1,000	5705
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C90402	1	-1,000	-1,000	8295
http://sweet.jpl.nasa.gov/2.3/humanAgriculture.owl#Horticulture	1	-1,000	-1,000	6196
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C13325	1	-1,000	-1,000	6215
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C85504	1	-1,000	-1,000	6800
http://purl.bioontology.org/ontology/MESH/D005838	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D059010	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/NCBITaxon_3701	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C63495	1	-1,000	-1,000	6806
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C42790	1	-1,000	-1,000	2313
http://cerrado.linkeddata.es/ecology/ccon#ComunityDynamics	1	-1,000	-1,000	-1
http://pubs.acs.org/doi/abs/10.1021/ac2001803	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/HP_0001945	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C3037	1	-1,000	-1,000	10604
http://purl.obolibrary.org/obo/CHEBI_60004	1	-1,000	-1,000	-1
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=7e182921-2b1c-4276-9495-2505417eb21d	1	-1,000	-1,000	9499
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C688	1	-1,000	-1,000	7495
http://edamontology.org/topic_3172	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C73427	1	-1,000	-1,000	7811
http://purl.obolibrary.org/obo/NCBITaxon_3039	1	-1,000	-1,000	-1
http://www.projecthalo.com/aura#Opportunistic-Pathogen	1	-1,000	-1,000	3803
http://purl.org/obo/owl/GO#GO_0042710	1	-1,000	-1,000	5803
http://purl.bioontology.org/ontology/MESH/D055432	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_24913	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/UBERON_0001088	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/SNOMEDCT/355001	1	-1,000	-1,000	-1

Annotation	Used Count	Meta Spec	Proc Spec	Run Time
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=cf0f592d-7883-4134-916f-3af8f0ad1c98	1	-1,000	-1,000	7211
http://genomics.uni-regensburg.de/software/NMR/MetaboQuant_1.2.zip	1	-1,000	-1,000	-1
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=9b73dac8-2486-4faf-be88-edeca0a97ed5	1	-1,000	-1,000	6085
http://www.ncbi.nlm.nih.gov/pmc/articles/PMC3426673/pdf/216_2012_Article_6226.pdf	1	-1,000	-1,000	6401
http://purl.bioontology.org/ontology/MESH/D014644	1	-1,000	-1,000	-1
http://mimi.case.edu/ontologies/2009/1/UnitsOntology#events_per_hour	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/CSP/0963-9394	1	-1,000	-1,000	-1
http://link.springer.com/article/10.1016%2Fj.jasms.2009.02.001	1	-1,000	-1,000	1616
http://ontology.neuinfo.org/NIF/DigitalEntities/NIF-Investigation.owl#birnlex_2072	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_10036	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D000758	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/NCBITAXON/1280	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/NCBIGene_839277	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D060434	1	-1,000	-1,000	-1
http://cran.r-project.org/web/packages/affy	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C802	1	-1,000	-1,000	7097
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C16237	1	-1,000	-1,000	3800
http://www.acdlabs.com/products/adh/nmr/1d_man/	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MEDDRA/10052360	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D002244	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D054884	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_16247	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MEDDRA/10058108	1	-1,000	-1,000	-1
http://www.ncbi.nlm.nih.gov/pubmed/21466230	1	-1,000	-1,000	8798
http://purl.obolibrary.org/obo/CHEBI_39027	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D056265	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_39026	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_39025	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_33284	1	-1,000	-1,000	-1
http://www.projecthalo.com/aura#Experimental-Group	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C20085	1	-1,000	-1,000	7802
http://purl.bioontology.org/ontology/MESH/D031204	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/CSP/2000-3145	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C104504	1	-1,000	-1,000	7698
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C25189	1	-1,000	-1,000	2199
http://purl.obolibrary.org/obo/SOY_0001692	1	-1,000	-1,000	-1
http://cerrado.linkeddata.es/ecology/ccon#SpeciesDiversity	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C3199	1	-1,000	-1,000	3101
http://purl.obolibrary.org/obo/DOID_1612	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D059305	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MSH/D020500	1	-1,000	-1,000	900
http://purl.obolibrary.org/obo/PW_0000640	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/SNOMEDCT/63158009	1	-1,000	-1,000	-1
http://purl.jp/bio/11/csso/CSSO_000119	1	-1,000	-1,000	-1
http://www.owl-ontologies.com/unnamed.owl#Commensal_species	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_62913	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/NCBITAXON/1131	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C66822	1	-1,000	-1,000	4500
http://www.projecthalo.com/aura#Blood-Brain-Barrier	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MSH/D003116	1	-1,000	-1,000	801
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C17895	1	-1,000	-1,000	3499
http://bioontology.org/projects/ontologies/birnlex#birnlex_2023	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/DOID_4	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C39564	1	-1,000	-1,000	4040
http://purl.obolibrary.org/obo/CHEBI_15929	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C16209	1	-1,000	-1,000	7700
http://nmrML.org/nmrCV#NMR:1400183	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C25284	1	-1,000	-1,000	5201
http://www.co-ode.org/ontologies/amino-acid/2006/05/18/amino-acid.owl#R	1	-1,000	-1,000	-1
http://scai.fraunhofer.de/CSEO#Modified_Risk_Tobacco_Product	1	-1,000	-1,000	-1
http://purl.org/obo/owl/PATO#PATO_0001034	1	-1,000	-1,000	801
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C15220	1	-1,000	-1,000	3701
http://omics.georgetown.edu/metabosearch.html	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/NCBITaxon_1718	1	-1,000	-1,000	-1
http://www.sciencedirect.com/science/article/pii/S0003269707006471	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C94604	1	-1,000	-1,000	9505
http://purl.obolibrary.org/obo/CHEBI_31746	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C94729	1	-1,000	-1,000	4301
http://purl.org/obo/owl/GO#GO_0044399	1	-1,000	-1,000	6799
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C36185	1	-1,000	-1,000	9205
http://purl.bioontology.org/ontology/CSP/2323-0799	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C41185	1	-1,000	-1,000	4459
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C43366	1	-1,000	-1,000	10402
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C14286	1	-1,000	-1,000	2797
http://purl.obolibrary.org/obo/NCBIGene_10057	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C16342	1	-1,000	-1,000	5639
http://purl.bioontology.org/ontology/NCBITAXON/120961	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C17557	1	-1,000	-1,000	4602
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C88198	1	-1,000	-1,000	3197

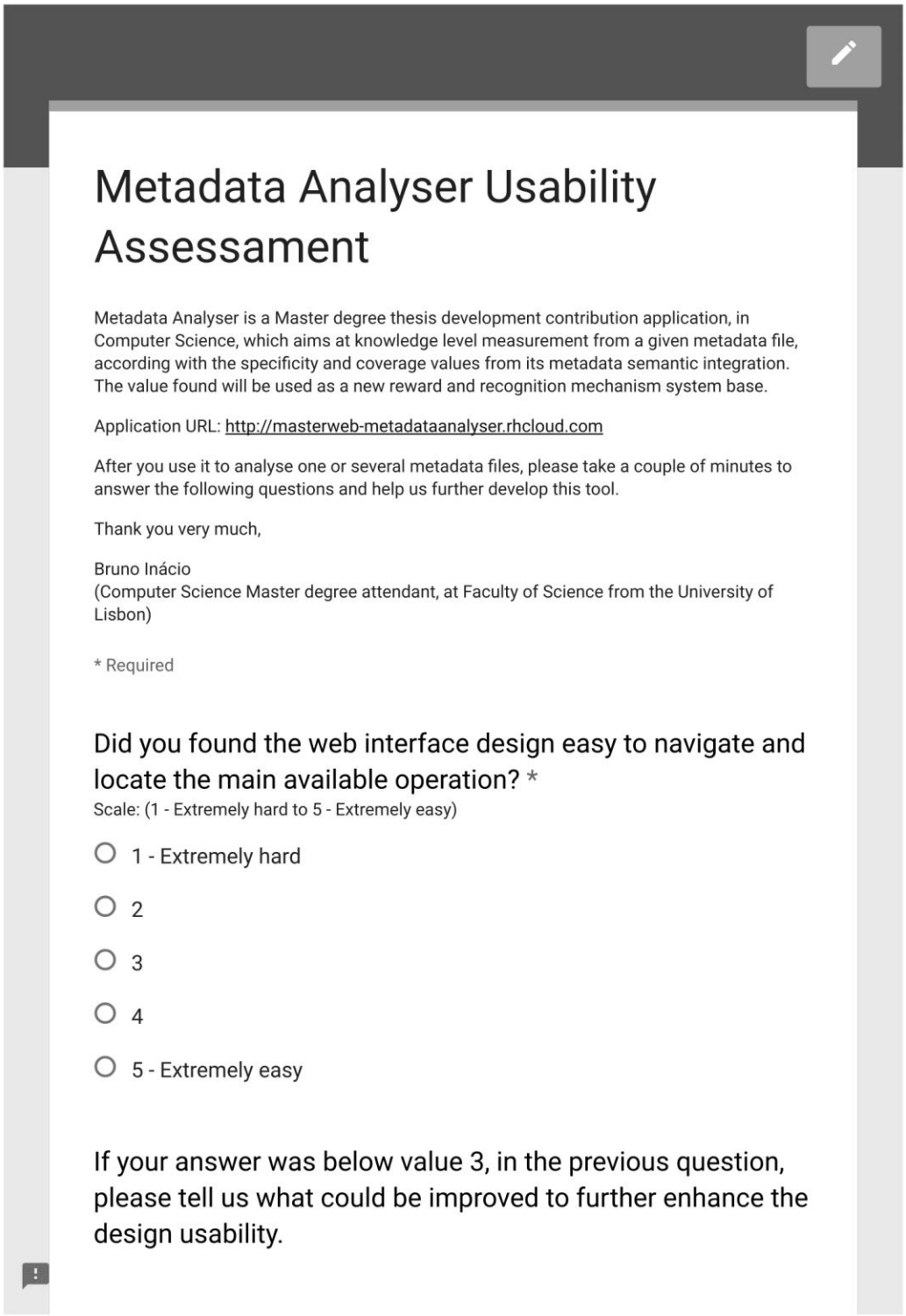
Annotation	Used Count	Meta Spec	Proc Spec	Run Time
http://purl.bioontology.org/ontology/MESH/D050155	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_59163	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/IDO_0000538	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/STY/T024	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C16212	1	-1,000	-1,000	4002
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C34538	1	-1,000	-1,000	3401
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=0f064dc0-075b-49da-b78c-0f67eec0a516	1	-1,000	-1,000	3703
http://edamontology.org/topic_0079	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C84399	1	-1,000	-1,000	7243
http://purl.bioontology.org/ontology/MSH/C081695	1	-1,000	-1,000	800
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C48938	1	-1,000	-1,000	5211
http://purl.obolibrary.org/obo/DOID_1324	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D061353	1	-1,000	-1,000	-1
http://pubs.acs.org/doi/suppl/10.1021/ac300829f/suppl_file/ac300829f_si_001.pdf	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C18143	1	-1,000	-1,000	6606
http://pubs.acs.org/doi/suppl/10.1021/ac300829f	1	-1,000	-1,000	-1
http://www.owl-ontologies.com/unnamed.owl#Keystone_species	1	-1,000	-1,000	-1
http://www.mcisb.org/]	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C25486	1	-1,000	-1,000	5304
http://edamontology.org/operation_3215	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/NCBITaxon_223283	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C17049	1	-1,000	-1,000	6900
http://purl.bioontology.org/ontology/MESH/D011786	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_35189	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_17855	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C24044	1	-1,000	-1,000	5501
http://mzmatch.sourceforge.net/	1	-1,000	-1,000	1198
http://purl.obolibrary.org/obo/TO_0000276	1	-1,000	-1,000	-1
http://www.co-ode.org/ontologies/galen#days	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C39095	1	-1,000	-1,000	8304
http://purl.obolibrary.org/obo/UBERON_0001969	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_67197	1	-1,000	-1,000	-1
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=747b1c79-3ef5-486f-b5ba-4faf0b0f55e5	1	-1,000	-1,000	3392
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C17632	1	-1,000	-1,000	8906
http://purl.obolibrary.org/obo/NCBITaxon_196627	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/SNOMEDCT/389086002	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/SNOMEDCT/50136005	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_53387	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/CHEBI_41609	1	-1,000	-1,000	-1
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=6d1d1929-3edf-4cb2-a134-78d2bc1aa0c2	1	-1,000	-1,000	7639
http://www.ifomis.org/acgt/1.0#Chemotherapy	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C17741	1	-1,000	-1,000	5897
http://purl.obolibrary.org/obo/HP_0001919	1	-1,000	-1,000	-1
http://purl.bioontology.org/ontology/MESH/D014157	1	-1,000	-1,000	-1
http://gmd.mpimp-golm.mpg.de/profile/default.aspx?XemlId=5f135f69-aff6-4d0a-8f7a-2f54be22d3e9	1	-1,000	-1,000	11498
http://purl.bioontology.org/ontology/MSH/D052817	1	-1,000	-1,000	1201
http://purl.obolibrary.org/obo/NCBITaxon_4565	1	-1,000	-1,000	-1
http://purl.obolibrary.org/obo/DRON_00723553	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#Bronchoalveolar_Lavage_Fluid	1	-1,000	-1,000	7299
http://purl.bioontology.org/ontology/MESH/D004249	1	-1,000	-1,000	-1
http://ncicb.nci.nih.gov/xml/owl/EVS/Thesaurus.owl#C18270	1	-1,000	-1,000	8000

Anexo N Implementação SQL do procedimento *sp_conceptspec*

```
CREATE PROCEDURE `sp_conceptspec`(IN  concept_iri VARCHAR(100), OUT
spec_value NUMERIC(6, 4))
READS SQL DATA BEGIN
   DECLARE start_time, end_time                        BIGINT;
   DECLARE owl_obj_id                                  INTEGER DEFAULT 0;
   DECLARE concept_ancestors_count                     INTEGER DEFAULT 0;
   DECLARE leaf_descendents_count                      INTEGER DEFAULT 0;
   DECLARE leaf_descendents_ancestors_count            INTEGER DEFAULT 0;
   DECLARE leaf_ancestors_count                        INTEGER DEFAULT 0;
   DECLARE leaft_concept_delta_sum                     INTEGER DEFAULT 0;
   -- Declare _val variables to read in each record from the cursor
   DECLARE subclass_val                                INTEGER;
   DECLARE name_val                                   TEXT;
   DECLARE distance_val                                INTEGER;
   DECLARE no_more_rows                                BOOLEAN DEFAULT FALSE;
   DECLARE loop_cntr                                   INTEGER DEFAULT 0;
   DECLARE
   leaf_descendents_cursor CURSOR FOR SELECT h.subclass, n.name, h.distance
             FROM hierarchy h INNER JOIN owl_objects o ON h.subclass = o.id
             INNER JOIN names n ON h.subclass = n.id
             INNER JOIN leaves l ON h.subclass = l.id
             WHERE o.type = 'Class' AND h.superclass =
              (SELECT w.id FROM owl_objects w
                    WHERE w.type = 'Class' AND LOWER(w.iri) = LOWER(
                          concept_iri)) h.superclass <> h.subclass
               ORDER BY h.distance ASC, h.subclass;
   -- declare handlers for exceptions
   DECLARE CONTINUE HANDLER FOR NOT FOUND SET no_more_rows = TRUE;
   -- read the starting time, in miliseconds
   SET start_time = UNIX_TIMESTAMP();
   -- get owl object id for the given concept iri
   SELECT f_get_owlid_from_iri(concept_iri)
     INTO owl_obj_id;
   -- check for a valid owl object
   IF owl_obj_id > 0
   THEN
      -- get all ancestors count for the given concept
      SELECT f_concept_ancestors_count(owl_obj_id)
         INTO concept_ancestors_count;
      -- check for worst case scenario, concept is a top ontology node
      IF concept_ancestors_count > 0 THEN
         -- get leaf descendents average calculus
         OPEN leaf_descendents_cursor;
         SELECT FOUND_ROWS() INTO leaf_descendents_count;
         -- check for best case scenario, concept is a leaf ontology node
         IF leaf_descendents_count > 0 THEN
            SET leaft_concept_delta_sum = 0;
           leaf_loop:
            LOOP
               FETCH leaf_descendents_cursor
                  INTO subclass_val, name_val, distance_val;
               -- break loop condition
               IF no_more_rows THEN
```

```
                CLOSE leaf_descendents_cursor;
                LEAVE leaf_loop;
             END IF;
             -- get ancestors count for this leaf subclass concept
             SELECT f_concept_ancestors_count(subclass_val)
                INTO leaf_ancestors_count;
             -- calculate leaf to concept delta distance
             SET leaft_concept_delta_sum =
                 leaft_concept_delta_sum +
                (leaf_ancestors_count - concept_ancestors_count);
             -- count the number of times looped
             SET loop_cntr = loop_cntr + 1;
          END LOOP leaf_loop;
          IF leaft_concept_delta_sum > 0 THEN
             -- calcule specification metric for the given concept
             SET spec_value = (concept_ancestors_count /
               (concept_ancestors_count
                       + (leaft_concept_delta_sum / loop_cntr)));
          ELSE
             -- something went wrong in delta calculus
             SET spec_value = 0;
          END IF;
       ELSE
          CLOSE leaf_descendents_cursor;
          -- this is a leaf concept, set the highest spec value
          SET spec_value = 1;
       END IF;
    ELSE
       -- this is a top concept in the ontology hierarchy
       SET spec_value = 0;
    END IF;
 ELSE
    -- this is not a valid concept, set the least spec value
    SET spec_value = -1;
 END IF;
END
```

Anexo O Questionário de usabilidade da solução desenvolvida



Metadata Analyser Usability Assessament

Metadata Analyser is a Master degree thesis development contribution application, in Computer Science, which aims at knowledge level measurement from a given metadata file, according with the specificity and coverage values from its metadata semantic integration. The value found will be used as a new reward and recognition mechanism system base.

Application URL: http://masterweb-metadataanalyser.rhcloud.com

After you use it to analyse one or several metadata files, please take a couple of minutes to answer the following questions and help us further develop this tool.

Thank you very much,

Bruno Inácio
(Computer Science Master degree attendant, at Faculty of Science from the University of Lisbon)

* Required

Did you found the web interface design easy to navigate and locate the main available operation? *
Scale: (1 - Extremely hard to 5 - Extremely easy)

- ○ 1 - Extremely hard
- ○ 2
- ○ 3
- ○ 4
- ○ 5 - Extremely easy

If your answer was below value 3, in the previous question, please tell us what could be improved to further enhance the design usability.

Your answer

Did you found it difficult to understand which option to take, in the analyser form, concerning the metadata file input options available? *

- Yes I did
- No I didn't

If you answered "Yes I did" in the previous question, please tell us what could be done to further simplify the metadata input method, in your opinion?

Your answer

Is there any other metadata file input method you think should or could be present in the analyser form?

Your answer

Was the analyser process progress bar, displayed after form submit, useful in understating the steps involved in the process of metadata file analysis? *

- Yes
- No

If you answered “No” in the previous question, please tell us how could the end user be informed about the analysis progress?

Your answer

Concerning the amount of time elapsed until a result was showed, did you found it reasonable regarding the metadata file complexity? *

- ○ Yes
- ○ No

If you answered "No" in the previous question, please tell us what is the average amount of time an analysis of this kind should take (in seconds).

Your answer

Did you found the metadata file analysis results presentation easy to find and understand? *

- ○ Yes I did
- ○ No I didn't

If you answered "No I didn't" in the previous question, please tell us how could the results be presented to the end user.

Your answer

Were the results, in a general form, what you expected them to be, regarding the submitted metadata file? *

- ○ Yes
- ○ No

If your answered "No" in the previous question, please tell us what was wrong or missing in your opinion.

Your answer

Were the study, classes and annotations specificity values coherent with what you expected? *

- Yes
- No

If you answered "No" in the previous question, please tell us what were the wrong values.

Your answer

Were the study and classes coverage values coherent with what you expected? *

- Yes
- No

If you answered "No" in the previous question, please tell us what were the wrong values.

Your answer

Please tell us about your overall opinion about this tool.

Your answer

100%: You made it.

SUBMIT

Never submit passwords through Google Forms.